infrastructures

MDPI

Article

Automated Classification of Exchange Information Requirements for Construction Projects Using Word2Vec and SVM

Ewelina Mitera-Kiełbasa * and Krzysztof Zima

Division of Management in Civil Engineering, Faculty of Civil Engineering, Cracow University of Technology, 31-155 Kraków, Poland; krzysztof.zima@pk.edu.pl

* Correspondence: e.mitera@pk.edu.pl

Abstract: This study addresses the challenge of automating the creation of Exchange Information Requirements (EIRs) for construction projects using Building Information Modelling (BIM) and Digital Twins, as specified in the ISO 19650 standard. This paper focuses on automating the classification of EIR paragraphs according to the ISO 19650 standard's categories, aiming to improve information management in construction projects. It addresses a gap in applying AI to enhance BIM project management, where barriers often include technological limitations, a shortage of specialists, and limited understanding of the methodology. The proposed method uses Word2Vec for text vectorisation and Support Vector Machines (SVMs) with an RBF kernel for text classification, and it attempts to apply Word2Vec with cosine similarity for text generation. The model achieved an average F1 score of 0.7, with predicted categories for provided sentences and similar matches for selected phrases. While the text classification results were promising, further refinement is required for the text generation component. This study concludes that integrating AI tools such as Word2Vec and SVM offers a feasible solution for enhancing EIR creation. However, further development of text generation, particularly using advanced techniques such as GPT, is recommended. These findings contribute to improving managing complex construction projects and advancing digitalization in the AECO sector.

Keywords: EIR; Word2Vec; SVM; cosine similarity; AI; BIM; Digital Twin; AECO; text classification; text generation

check for updates

Citation: Mitera-Kiełbasa, E.; Zima, K. Automated Classification of Exchange Information Requirements for Construction Projects Using Word2Vec and SVM. *Infrastructures* **2024**, *9*, 194. https://doi.org/10.3390/infrastructures9110194

Academic Editor: Andrzej Borkowski

Received: 19 September 2024
Revised: 17 October 2024
Accepted: 21 October 2024
Published: 29 October 2024

1. Introduction

The Architecture, Engineering, Construction, and Operation (AECO) sector is experiencing a significant digital transformation in alignment with the European Commission's *Industry 5.0* proposal, which promotes a digital and green transition [1]. There is a growing emphasis on the implementation of Building Information Modelling (BIM) and Digital Twins to enhance the efficiency of both public and private construction projects. These methodologies support the European Union's goal of fostering digitalization within the industry, and some countries (35% of European countries [2]) have committed to their use for public construction projects to varying degrees.

Building Information Modelling (BIM) facilitates the creation of a shared digital representation of built assets, which streamlines design, construction, and operational processes [3]. In the literature, BIM refers to both the methodology and the models themselves [4]. However, it is more accurate to speak of „models" in the plural, as in practice, each discipline typically operates using its own model. This methodology outlines the process for creating, storing, and exchanging information on building assets, and aims at interoperability. The Digital Twin (DT) extends this by integrating real-time data [5] from interconnected devices, such as sensors and Internet of Things (IoT) applications, enabling advanced simulations and AI-driven analyses [6]. DT plays a crucial role in intelligent and smart construction [7]. BIM is supported by the ISO 19650 international standard, which standardizes information management and facilitates experience sharing across projects. Despite these advantages, managing a construction project using BIM remains complex, requiring specialized knowledge and strategic planning for the entire life cycle of a built asset. Technological barriers, a lack of expertise, and insufficient understanding of the methodology are often cited as challenges to BIM implementation [8,9]. The ISO 19650 standard underscores the importance of engaging expert knowledge in such processes [10].

This study addresses these challenges by proposing an automated, personalized approach to creating Exchange Information Requirements (EIRs), which are crucial for outlining the client's (project owner's, general contractor's, or designer's) requirements for the contractor (designer or subcontractor). Here, automation focuses on classifying EIR paragraphs according to the ISO 19650 standard's managerial, commercial, and technical categories, contributing to more efficient information management in BIM/Digital Twin projects. AI integration offers a viable solution to these challenges, streamlining the process and facilitating the adoption of these methodologies. This is particularly relevant to the European Union's digital transition in the AECO sector.

While existing research highlights the potential of AI in the construction sector, gaps remain in its application to automated information management. Gohel et al. reviewed AI's role in construction management [11], and Rangasamy et al. examined the integration of BIM and AI in prefabricated construction [12]. Some studies have proposed improvements through the use of classification algorithms, such as that of Ma et al., which uses rule-based methods for diagnosing faults in urban rail systems [13]. Zheng et al. developed a domain-specific language model for the AEC sector, improving text classification and entity recognition [14]. Dolhopolov et al. utilized AI in analyzing Digital Twins of buildings and construction sites [15]. Advanced classification models have been introduced outside the AECO sector, such as Bartal et al.'s ADA model for narrative classification to effectively detect post-traumatic stress disorder following childbirth [16] and Huang et al.'s FinBERT language model, which is adapted to the financial sector [17]. However, despite these advancements, the automation of information requirements (IRs) in the BIM/Digital Twin context has received insufficient attention.

Further studies, such as those by Piazzi et al., Goonetillake et al., and Tomczak et al., have focused on improving the specification and exchange of information within BIM [18–20]. Piazzi et al. focused on graphical Exchange Information Requirements, using advanced algorithms and rule-based systems to enhance data exchange efficiency and accuracy [18]. Goonetillake et al. developed a prototype tool for embedding digital Exchange Information Requirements, enhancing the automation and accuracy of capturing as-built information through process maps and sample information requirements [19]. Tomczak et al. presented a comprehensive review of methods used to specify information requirements in digital construction projects, comparing standardized and non-standardized approaches to improve efficiency and communication in BIM environments [20].

Text generation research, which attempts to create text sequences based on existing documents, dates back to the 1950s. Luhn described an algorithm for automatically summarizing documents [21], and today, techniques such as abstractive summarisation and deep learning models are employed [22,23]. However, in the early stages of such research, simpler methods like semantic similarity searches can be used [24,25].

The objective of this research was to develop the automatic categorization of data from a database of published Exchange Information Requirements (EIRs) in both public and private tenders and bidding procedures conducted using BIM or Digital Twin methodologies. This categorization aligns with the ISO 19650 international standard, with the potential future integration of a chatbot. This study addresses the challenge of automating EIR creation, streamlining the information management process in BIM/Digital Twins, and responding to the need for an easier implementation of these methodologies. Furthermore, it aligns with the European Union's digital transition goals within the AECO sector. By addressing this gap, this study not only advances the application of AI in construction project management but also contributes to the practical implementation of standardized methodologies, such as the ISO 19650 standard, thereby facilitating the smoother adoption of BIM and Digital Twins across the sector.

2. Materials and Methods

This paper reports on the development of an automated model for proposing Exchange Information Requirements (EIRs) tailored to user preferences and a specific construction project, implemented in BIM/as a Digital Twin, with focus text classification into the ISO 19650 standard. This section highlights the use of Word2Vec, Support Vector Machines (SVMs), and the Radial Basis Function (RBF) kernel for text classification in accordance with the ISO 19650 standard. A future paper will also discuss the application of the Term Frequency–Inverse Document Frequency (TF-IDF) statistical method for the numerical representation of text, as well as Bidirectional Encoder Representations from Transformers (BERT) for vectorization and classification, based on transformer architectures. Figure 1 illustrates the overall methodology, with the section relevant to this paper highlighted in red.

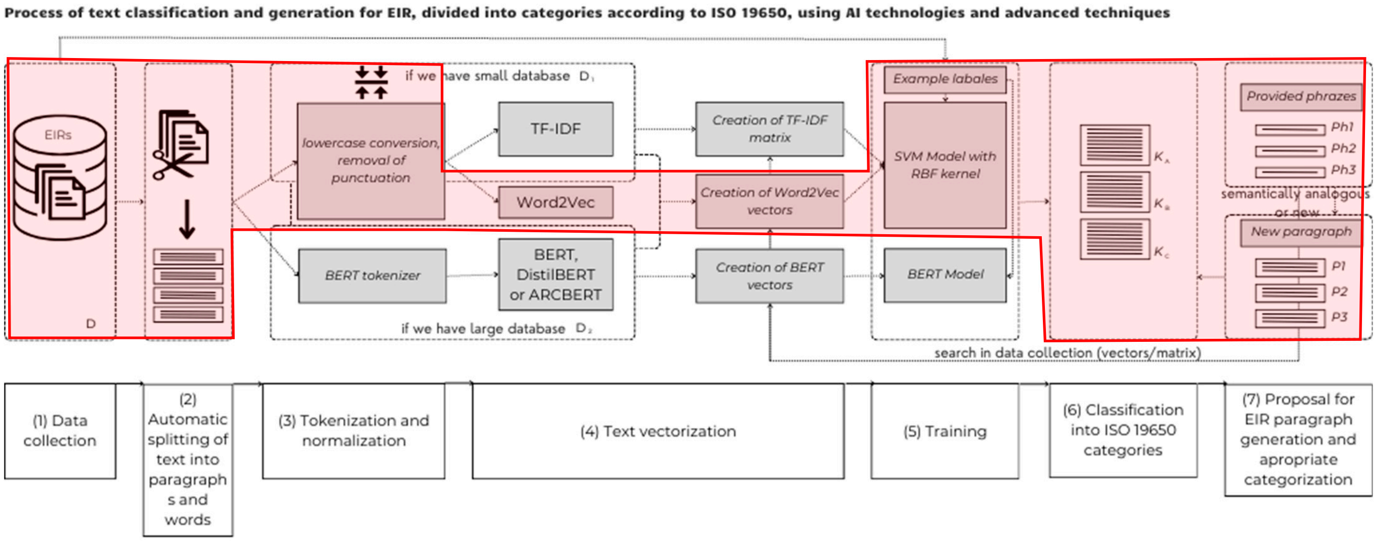

Process of text classification and generation for EIR, divided into categories according to ISO 19650, using AI technologies and advanced techniques
EIRs
D
if we have small database D₁
lowercase conversion, removal of punctuation
TF-IDF
Word2Vec
BERT tokenizer
BERT, DistilBERT or ARCBERT
if we have large database D₂
Creation of TF-IDF matrix
Creation of Word2Vec vectors
Creation of BERT vectors
Example labales
SVM Model with RBF kernel
BERT Model
K_A
K_B
K_C
Provided phrazes
Ph1
Ph2
Ph3
semantically analogous or new
New paragraph
P1
P2
P3
search in data collection (vectors/matrix)
(1) Data collection
(2) Automatic splitting of text into paragraph s and words
(3) Tokenization and normalization
(4) Text vectorization
(5) Training
(6) Classification into ISO 19650 categories
(7) Proposal for EIR paragraph generation and apropriate categorization

Figure 1. Process of text classification and generation for the EIR proposal presented (in red using Word2Vec and SVM).

2.1. Step (1): Creation of Data Collection

The dataset required for training the model comprises Exchange Information Requirement (EIR) documents from both public and private tenders. The user will also be able to upload their own documents, for instance, from previous projects, allowing for the refinement of established patterns. Several countries publish EIR templates, which provide an appropriate foundation for this process. It is important to note that these documents do not necessarily need to be categorized according to the ISO 19650 standard, as the model is designed to automatically assign paragraphs to the relevant categories.

2.2. Steps (2) and (3): Text Preprocessing

The collected data must be adequately prepared for text vectorization, which requires text preprocessing. Since the aim is to use fragments of EIRs for a proposal form, tailored to user preferences, the text is first divided into paragraphs (step 2), followed by normalization and tokenization, as it is from tokens that feature vectors are created. The text undergoes normalization, meaning letters are converted to lowercase, and special characters and whitespace, such as spaces, are removed. The text is subsequently tokenized, which involves splitting the text into individual words. For example, the sentence "The contractor shall compile the BEP." would be tokenized using a Natural Language Toolkit (NLTK)-

based script as: [“The”, “contractor”, “shall”, “compile”, “the”, “BEP”, “.”]. This structured approach ensures that the data are prepared for efficient processing in the subsequent stages of classification and categorization.

2.3. Step (4): Text Vectorization Using Word2Vec

To achieve text categorization, the text must first be converted into numerical values, specifically creating numerical feature vectors for paragraphs. In this case, the feature vector for a paragraph is the arithmetic mean of the feature vectors of the tokens within that paragraph, generated during the preprocessing phase. The Word2Vec method was selected for this, using static word embedding, meaning static vector representation. Some methods require larger datasets for training, while others can function effectively with smaller datasets. Therefore, it was decided to explore the implementation of text vectorization using TF-IDF, which performs well with smaller datasets (in this case, EIRs), and BERT, which yields better results with larger datasets (as shown in Figure 1). This paper focuses on the use of Word2Vec, which can be applied to large datasets, but due to its pre-trained nature on a vast corpus, it should also perform adequately with smaller datasets.

Word2Vec falls under unsupervised learning methods, which do not require labelled data for training (labelling is applied during classification). While it uses neural networks, it employs shallow networks, with only one hidden layer of neurons.

One key advantage of this method is that the vectors generated possess semantic relationships, allowing for the identification of similar words. For example, “LOD” (Level of Detail or Level of Development in BIM models, as specified by the PAS 1192 standard and BIM Forum) and “LOIN” (Level of Information Need used in the ISO 19650 standard) can be compared by calculating the cosine of the angle between their vectors. Moreover, Word2Vec is pre-trained on a large corpus, enabling it to perform well even with limited training resources.

Two architectures can be employed: the Continuous Bag of Words (CBoW) architecture for smaller datasets, and the Skip-Gram architecture for larger ones [26], with the code in [27]. The difference between these architectures is illustrated in Figure 2. CBoW predicts a target word based on its surrounding context, while Skip-Gram predicts surrounding words based on a target word [26,28].

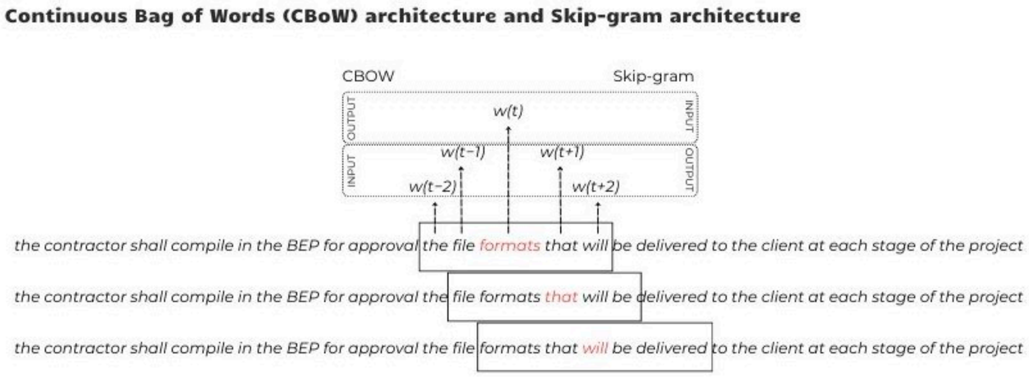

Continuous Bag of Words (CBoW) architecture and Skip-gram architecture
CBOW
Skip-gram
OUTPUT
INPUT
INPUT
OUTPUT
w(t)
w(t-1)
w(t+1)
w(t-2)
w(t+2)
the contractor shall compile in the BEP for approval the file formats that will be delivered to the client at each stage of the project
the contractor shall compile in the BEP for approval the file formats that will be delivered to the client at each stage of the project
the contractor shall compile in the BEP for approval the file formats that will be delivered to the client at each stage of the project

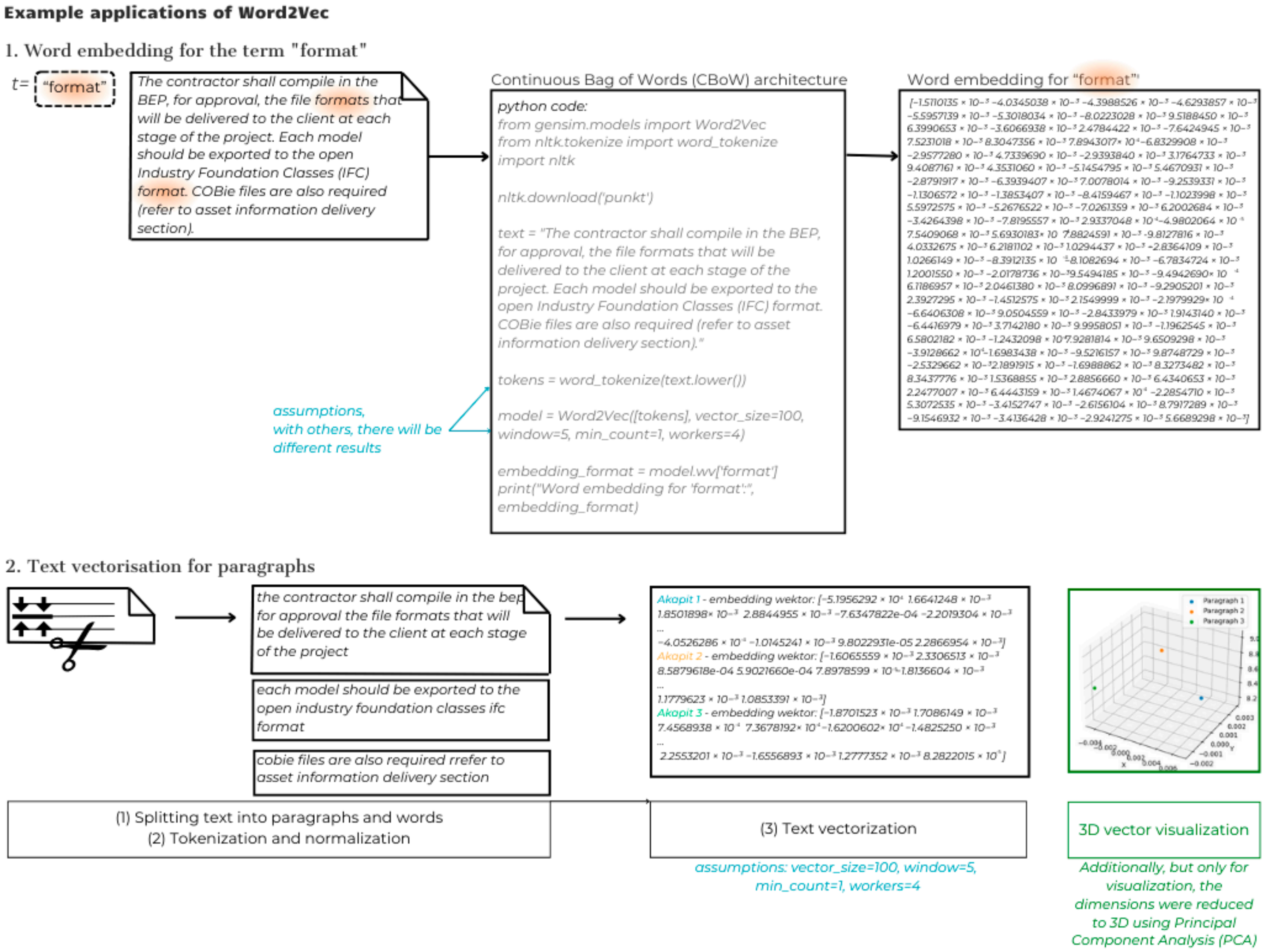
Figure 2. Differences between Continuous Bag of Words and Skip-Gram architectures.

The mathematical formulation of the objective function for the Skip-Gram model is shown in Equation (1) [28], which maximizes the average log probability, where $\{w_1, w_2, \ldots, w_T\}$ is the sequence of training words, c is the size of the training context, and w_T is the central word. The softmax function $p(w_O|w_I)$ is used, where v_w and v'_w are the “input” and “output” vector representations of w, and W is the number of words in the vocabulary.

$$SG = \frac{1}{T}\sum_{t=1}^{T}\sum_{-c\leq j\leq c, j\neq 0} \log\ p(w_{t+j}|w_t), \text{ where } p(w_O|\ w_I) = \frac{\exp\left({v'_{wO}}^{T} v_{wI}\right)}{\sum_{w=1}^{W} \exp\ ({v'_w}^{T} v_{wI})} \tag{1}$$

Mikolov noted that calculating the denominator of this probability can be computationally problematic, considering that it accounts for all words in the vocabulary. Thus, hierarchical softmax or the approximation of the softmax function by Noise-Contrastive Estimation (NCE), which can be simplified by negative sampling (NEG), was proposed.

Word embedding for the word "format" for a sample EIR fragment (i.e., creating domain-specific models) is illustrated in Figure 3. With different assumptions, such as a smaller vector size, the results will differ. The *gensim* library allows Word2Vec models to be trained on a custom dataset.


Example applications of Word2Vec
1. Word embedding for the term "format"
t= "format"
The contractor shall compile in the BEP, for approval, the file formats that will be delivered to the client at each stage of the project. Each model should be exported to the open Industry Foundation Classes (IFC) format. COBie files are also required (refer to asset information delivery section).
Continuous Bag of Words (CBoW) architecture
Word embedding for "format"
assumptions, with others, there will be different results
2. Text vectorisation for paragraphs
the contractor shall compile in the bep for approval the file formats that will be delivered to the client at each stage of the project
each model should be exported to the open industry foundation classes ifc format
cobie files are also required rrefer to asset information delivery section
(1) Splitting text into paragraphs and words
(2) Tokenization and normalization
(3) Text vectorization
assumptions: vector_size=100, window=5, min_count=1, workers=4
Paragraph 1
Paragraph 2
Paragraph 3
3D vector visualization
Additionally, but only for visualization, the dimensions were reduced to 3D using Principal Component Analysis (PCA)

```
python code:
from gensim.models import Word2Vec
from nltk.tokenize import word_tokenize
import nltk

nltk.download('punkt')

text = "The contractor shall compile in the BEP,
for approval, the file formats that will be
delivered to the client at each stage of the
project. Each model should be exported to the
open Industry Foundation Classes (IFC) format.
COBie files are also required (refer to asset
information delivery section)."

tokens = word_tokenize(text.lower())

model = Word2Vec([tokens], vector_size=100,
window=5, min_count=1, workers=4)

embedding_format = model.wv['format']
print("Word embedding for 'format':",
embedding_format)
```

```
[−1.5110135 × 10−3 −4.0345038 × 10−3 −4.3988526 × 10−3 −4.6293857 × 10−3
−5.5957139 × 10−3 −5.3018034 × 10−3 −8.0223028 × 10−3 9.5188450 × 10−3
6.3990653 × 10−3 −3.6066938 × 10−3 2.4784422 × 10−3 −7.6424945 × 10−3
7.5231018 × 10−3 8.3047356 × 10−3 7.8943017× 10−4 −6.8329908 × 10−3
−2.9577280 × 10−3 4.7339690 × 10−3 −2.9393840 × 10−3 3.1764733 × 10−3
9.4087161 × 10−3 4.3531060 × 10−3 −5.1454795 × 10−3 5.4670931 × 10−3
−2.8791917 × 10−3 −6.3939407 × 10−3 7.0078014 × 10−3 −9.2539331 × 10−3
−1.1306572 × 10−3 −1.3853407 × 10−3 −8.4159467 × 10−3 −1.1023998 × 10−3
5.5972575 × 10−3 −5.2676522 × 10−3 −7.0261359 × 10−3 6.2002684 × 10−3
−3.4264398 × 10−3 −7.8195557 × 10−3 2.9337048 × 10−4 −4.9802064 × 10−4
7.5409068 × 10−3 5.6930183× 10−4 7.8824591 × 10−3 -9.8127816 × 10−3
4.0332675 × 10−3 6.2181102 × 10−3 1.0294437 × 10−3 −2.8364109 × 10−3
1.0266149 × 10−3 −8.3912135 × 10−4 8.1082694 × 10−3 −6.7834724 × 10−3
1.2001550 × 10−3 −2.0178736 × 10−3 9.5494185 × 10−3 −9.4942690× 10−4
6.1186957 × 10−3 2.0461380 × 10−3 8.0996891 × 10−3 −9.2905201 × 10−3
2.3927295 × 10−3 −1.4512575 × 10−3 2.1549999 × 10−3 −2.1979929× 10−4
−6.6406308 × 10−3 9.0504559 × 10−3 −2.8433979 × 10−3 1.9143140 × 10−3
−6.4416979 × 10−3 3.7142180 × 10−3 9.9958051 × 10−3 −1.1962545 × 10−3
6.5802182 × 10−3 −1.2432098 × 10−4 7.9281814 × 10−3 9.6509298 × 10−3
−3.9128662 × 10−4 −1.6983438 × 10−3 −9.5216157 × 10−3 9.8748729 × 10−3
−2.5329662 × 10−3 2.1891915 × 10−3 −1.6988862 × 10−3 8.3273482 × 10−3
8.3437776 × 10−3 1.5368855 × 10−3 2.8856660 × 10−3 6.4340653 × 10−3
2.2477007 × 10−3 6.4443159 × 10−3 1.4674067 × 10−4 −2.2854710 × 10−3
5.3072535 × 10−3 −3.4152747 × 10−3 −2.6156104 × 10−3 8.7917289 × 10−3
−9.1546932 × 10−3 −3.4136428 × 10−3 −2.9241275 × 10−3 5.6689298 × 10−3]
```

```
Akapit 1 - embedding wektor: [−5.1956292 × 10−4 1.6641248 × 10−3
1.8501898× 10−3 2.8844955 × 10−3 −7.6347822e-04 −2.2019304 × 10−3
...
−4.0526286 × 10−4 −1.0145241 × 10−3 9.8022931e-05 2.2866954 × 10−3]
Akapit 2 - embedding wektor: [−1.6065559 × 10−3 2.3306513 × 10−3
8.5879618e-04 5.9021660e-04 7.8978599 × 10−4 −1.8136604 × 10−3
...
1.1779623 × 10−3 1.0853391 × 10−3]
Akapit 3 - embedding wektor: [−1.8701523 × 10−3 1.7086149 × 10−3
7.4568938 × 10−4 7.3678192× 10−4 −1.6200602× 10−4 −1.4825250 × 10−3
...
2.2553201 × 10−3 −1.6556893 × 10−3 1.2777352 × 10−3 8.2822015 × 10−5]
```

Figure 3. Example applications of Word2Vec for text vectorization.

Text vectorization for paragraphs begins with splitting the text into paragraphs (schematically illustrated in Figure 3), followed by tokenization and normalization, and then text vectorization for each paragraph separately, where the average embedding vector for all words in the paragraph is calculated.

Improvements to the Word2Vec method include Global Vectors for Word Representation (GloVe) and FastText, which are suitable topics for future research.

2.4. Steps (5) and (6): Text Classification into the ISO 19650 Standard Categories

After preparing the text, the core of this study can be addressed: the classification of text into ISO 19650 categories. Before delving into the classification algorithms, it is essential to provide an overview of the Exchange Information Requirements (EIRs) as outlined in the ISO 19650 and explain the associated categories of information.

2.4.1. Exchange Information Requirements According to the ISO 19650 Standard

The ISO 19650 standard is an international standard for managing construction projects using BIM. According to this standard, an Exchange Information Requirements (EIR) document is a "specification for what, when, how, and for whom information is to be produced in relation to an appointment (agreed instruction for the provision of information concerning works, goods, or services)" [3]. This document is crucial in real-estate development process, as proper preparation aims to prevent dissatisfaction with the project outcome, as it clearly outlines the client's expectations [29]. In cases where specialized knowledge is lacking, it is advisable to consult experts in the field.

The contractor—whether it is a design firm tasked with producing comprehensive design documentation, a general contractor responsible for construction in a Design–Bid–Build system, or a subcontractor—has to submit a BIM Execution Plan for approval. This plan should meet the EIRs and provide more detailed project-specific information. It is essential to consider the entire life cycle of the building asset when specifying expectations, as a well-designed model can prove beneficial in later stages, such as during a building's operation, facilitating quicker access to equipment warranties or user manuals.

The ISO 19650 standard categorizes EIR information into three main groups: managerial, commercial, and technical. In the non-mandatory Polish BIM guidelines, managerial aspects include standards, roles and responsibilities, data security, coordination and clash detection, collaboration organization, meetings and model reviews, Health and Safety (H&S) management, and project management. Commercial aspects cover data exchange/delivery schedule, strategic goals, defining expectations regarding BIM scope, and competency assessment, while technical aspects involve technical or software platforms, data exchange formats, coordinates, levels of detail, and training [30].

For example, "roles and responsibilities" within managerial aspects might include a table that outlines who is responsible for what tasks in the project, ensuring clarity in accountability.

"Strategic goals" in the commercial domain might detail not only project-specific objectives but also the project owner's broader organizational goals. It is important to note that information requirements go beyond just EIRs. The ISO 19650 standard outlines six main types of information requirements: Organizational Information Requirements, Project Information Requirements, Asset Information Requirements, Exchange Information Requirements, and the deliverables associated with them: the Asset Information Model and the Project Information Model. This highlights the complexity of the issues that must be addressed to develop appropriate guidelines.

One example of technical requirements includes the Levels of Information Need, which are divided into geometrical, alphanumerical, and document-based information. Geometrical information pertains to dimensions, such as the size of windows or spans, while alphanumerical data specify materials, such as the concrete class to be used. It is crucial to ensure that the information provided is necessary for the projects, as excessive detail can lead to waste, such as overly detailed geometrical representations or unnecessary model parameters that are irrelevant for analysis or operation [31].

2.4.2. Support Vector Machine

For text classification into the ISO 19650 categories, the Support Vector Machine (SVM) method was applied. Text classification using BERT was also proposed for future studies, as outlined in Figure 1. However, this transformer-based model is more effective when applied to large datasets, which is why an alternative for smaller datasets was suggested. One key consideration was selecting a method capable of handling classification across three categories that may not be linearly separable in a multidimensional space (specifically, a 100-dimensional space in our case). Therefore, the Radial Basis Function (RBF) kernel was proposed for SVM, which is particularly suited to such scenarios, which will be explained further in the paper.

SVM is a supervised learning method used for classification, regression, and anomaly detection tasks. It operates by identifying a hyperplane or a set of hyperplanes in a multidimensional space to separate different classes. SVM is particularly effective in complex scenarios where data are not linearly separable and is capable of handling both binary and multi-class classification problems [32].

To clearly explain this method, one can consider a space with two sets of features separated by a linear hyperplane. The distance between the nearest points (*support vectors*) of these sets and the hyperplane defines the margin γ (Equation (3)). Maximizing this margin results in an optimal hyperplane. The hyperplane in an n-dimensional feature space is described by Equation (2) [33], where ω is the weight vector, x is the feature vector, and b is the bias term.

$$\omega \cdot x - b = 0 \tag{2}$$

$$\gamma = \frac{2}{\|\omega\|} \tag{3}$$

The margin is maximized by minimizing $\|\omega\|$.

As previously outlined, there are three categories (managerial, commercial, and technical), meaning a multi-class classification problem must be solved. In vector space, the groups of vectors that correspond to these categories may not be linearly separable. To address this, the Radial Basis Function (RBF) kernel was employed, which is particularly suited to non-linearly separable data. In practice, techniques such as One-vs-Rest or One-vs-One are often used. In One-vs-Rest, the model first distinguishes K_A from K_B and K_C, and so on. In or One-vs-One, the model differentiates K_A from K_B initially.

The RBF kernel, described by the kernel function in Equation (4) (where x_i and x_j are feature vectors, and r is a parameter that defines the influence range of a single training sample) [34], maps the data into higher dimensions to make the classes linearly separable, utilizing the Kernel Trick.

$$K(x_i, x_j) = \exp\left(-r\|x_i - x_j\|^2\right) \tag{4}$$

This method is complex, particularly in the context of non-linear separability and multi-class complexity. Therefore, this article focuses only on its characteristic components.

An SVM classifies new data based on support vectors, with the decision function represented by Equation (5) [35], where NS refers to the number of support vectors, α_i are the weights assigned to the support vectors, y_i are class labels, x_i are support vectors, and b is the bias term.

$$f(x) = \sum_{i=1}^{NS} \alpha_i y_i K(x_i, x) + b \tag{5}$$

The objective function for an SVM with an RBF kernel aims to maximize the difference between the sum of the Lagrange multipliers α_i and α_j and half the weighted sum of the dot products of the training vectors x_i and x_j, which allows the optimal hyperplane that maximizes the margin between the classes to be found, as shown in Equation (6) [36].

$$max \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j K(x_i, x_j),\ \text{subject to} \sum_{i=1}^{N} \alpha_i y_i = 0,\ \alpha_i \geq 0 \tag{6}$$

where α_i and α_j are Lagrange multipliers, and y_i, y_j are class labels.

In practice, data labelling is conducted to enable the model to learn how to differentiate categories. Specific files are labelled as K_A, K_B, or K_C, corresponding to managerial, commercial, or technical aspects, respectively. As a result, when new EIR files are uploaded, the model can automatically classify paragraphs into the appropriate categories.

The performance of the trained model was evaluated using metrics such as accuracy, precision, recall, and F1 score. The scikit-learn library was utilized to calculate these parameters [37]. According to Equation (7), *accuracy* is the ratio of correct predictions to the total number of examples, $\hat{y}_i$ is the predicted value of the i-th sample, y_i is the actual value, $n_{samples}$ is the total number of samples, and $1(\hat{y}_i = y_i)$ is the indicator function, representing the number of correct predictions in the dataset.

$$accuracy(y,\ \hat{y}) = \frac{1}{n_{samples}} \sum_{i=0}^{n_{samples}-1} 1(\hat{y}_i = y_i) \tag{7}$$

Precision P (Equation (8)) measures the proportion of true positives T_p relative to the sum of true positives and false positives F_p. It indicates the percentage of instances that the model classified as positive which were actually positive. *Recall R* (Equation (9) also compares the number of true positives but relative to the sum of true positives and false negatives F_n, measuring the model's ability to capture all relevant positive examples.

$$\mathrm{P} = \frac{T_p}{T_p + F_p} \tag{8}$$

$$\mathrm{R} = \frac{T_p}{T_p + F_n} \tag{9}$$

For example, in a dataset of 50 paragraphs, where 12 belong to category K_A, if the model predicts that 10 paragraphs fall under K_A and 8 of these are correctly classified (i.e., T_p = 8), while 2 are incorrectly classified as K_A (i.e., F_p = 2), and 4 paragraphs that belong to K_A are not identified by the model (i.e., F_n = 4), then the precision P = 8/10 = 0.8 (80%) and the recall R = 8/12 = 0.67R = 8/12 = 0.67 (67%).

The *F1 score* is the harmonic mean of precision and recall, providing a balanced measure between the two, particularly when there is a disparity between positive and false predictions, according to Equation (10).

$$\mathrm{F1} = \frac{2 * T_p}{2 * T_p + F_p + F_n} \tag{10}$$

An analysis of text classification was performed for selected sentences to validate the model's effectiveness.

2.5. Step (7): Proposal for EIR Paragraph Generation and Appropriate Categorization

Although the text classification process could be considered complete at this point, it is the foundation for the subsequent step. This next phase involves the specification of the client's preferences through the use of a chatbot, which would assign appropriate sentences or paragraphs of a proposed EIR. This phase will be refined in the future as part of the development of an automatic, personalized EIR generation methodology, where a Generative Pre-trained Transformer (GPT), based on the Transformer architecture, will also be employed.

In this study, however, the performance of the existing model in text generation was preliminarily tested. Example phrases were provided, and the model was tasked with finding similar phrases by calculating the cosine similarity between the vector of the input phrase and the paragraph vectors in the database.

The *cosine similarity k(x,y)* (Equation (11)) between two vectors x and y is calculated as the ratio of the dot product of vectors x and y, with n-dimensions, to the product of the norms of x and y *(*$\|x\|$*,* $\|y\|$*)* [37]. In practice, it computes the cosine of the angle between these vectors in a multidimensional space. A result close to 1 indicates high similarity, while a result close to -1 signifies that the vectors are oriented in opposite directions.

$$k(x,y) = \frac{x{\cdot}y\prime}{\|x\|\|y\|},\ \text{where } x{\cdot}y\prime = \sum_{i=1}^{n} x_i{\cdot}y_i,\ \|x\| = \sqrt{\sum_{i=1}^{n} x_i^2}, \|y\| = \sqrt{\sum_{i=1}^{n} y_i^2} \tag{11}$$

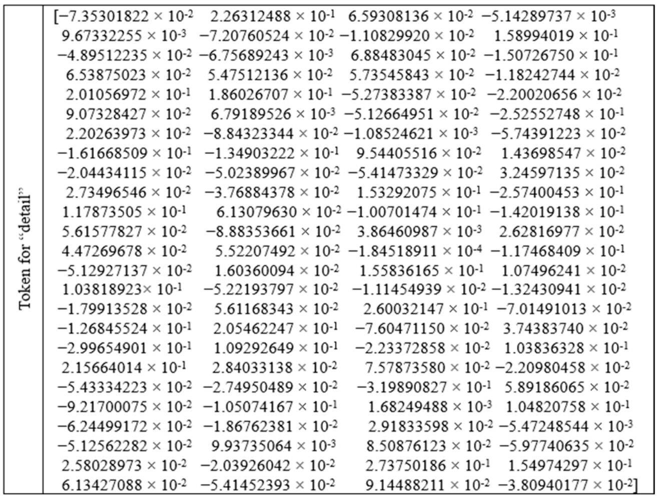
If the model does not find a suitable match, it generates a new sentence containing the given phrase. These generated sentences are then categorized into the appropriate K_A, K_B, or K_C (managerial, commercial, or technical aspects).

This step lays the groundwork for future research on the automatic generation of EIRs, enabling more personalized and precise output aligned with stakeholder preferences.

3. Results

The dataset used to validate the model comprised 8494 paragraphs categorized under K_A (managerial), 5130 under K_B (commercial), and 944 under K_C (technical). The training and validation data were split in the commonly used 80:20 ratio [38]. The varying quantity of data across categories may have influenced the results. The model was expected to be more accurate in classifying K_A due to the larger volume of training data available for this category.

Figure 4 illustrates the vector for the token "detail" generated through text vectorization. Feature vectors were set to 100 dimensions, a crucial aspect for capturing semantic relationships between words. This dimensionality is key for conducting the semantic analysis required for this task, which reflects the complexity of the methods employed.



Token for "detail"	[$-7.35301822 \times 10^{-2}$	$2.26312488 \times 10^{-1}$	$6.59308136 \times 10^{-2}$	$-5.14289737 \times 10^{-3}$
	$9.67332255 \times 10^{-3}$	$-7.20760524 \times 10^{-2}$	$-1.10829920 \times 10^{-2}$	$1.58994019 \times 10^{-1}$
	$-4.89512235 \times 10^{-2}$	$-6.75689243 \times 10^{-3}$	$6.88483045 \times 10^{-2}$	$-1.50726750 \times 10^{-1}$
	$6.53875023 \times 10^{-2}$	$5.47512136 \times 10^{-2}$	$5.73545843 \times 10^{-2}$	$-1.18242744 \times 10^{-2}$
	$2.01056972 \times 10^{-1}$	$1.86026707 \times 10^{-1}$	$-5.27383387 \times 10^{-2}$	$-2.20020656 \times 10^{-2}$
	$9.07328427 \times 10^{-2}$	$6.79189526 \times 10^{-3}$	$-5.12664951 \times 10^{-2}$	$-2.52552748 \times 10^{-1}$
	$2.20263973 \times 10^{-2}$	$-8.84323344 \times 10^{-2}$	$-1.08524621 \times 10^{-3}$	$-5.74391223 \times 10^{-2}$
	$-1.61668509 \times 10^{-1}$	$-1.34903222 \times 10^{-1}$	$9.54405516 \times 10^{-2}$	$1.43698547 \times 10^{-2}$
	$-2.04434115 \times 10^{-2}$	$-5.02389967 \times 10^{-2}$	$-5.41473329 \times 10^{-2}$	$3.24597135 \times 10^{-2}$
	$2.73496546 \times 10^{-2}$	$-3.76884378 \times 10^{-2}$	$1.53292075 \times 10^{-1}$	$-2.57400453 \times 10^{-1}$
	$1.17873505 \times 10^{-1}$	$6.13079630 \times 10^{-2}$	$-1.00701474 \times 10^{-1}$	$-1.42019138 \times 10^{-1}$
	$5.61577827 \times 10^{-2}$	$-8.88353661 \times 10^{-2}$	$3.86460987 \times 10^{-3}$	$2.62816977 \times 10^{-2}$
	$4.47269678 \times 10^{-2}$	$5.52207492 \times 10^{-2}$	$-1.84518911 \times 10^{-4}$	$-1.17468409 \times 10^{-1}$
	$-5.12927137 \times 10^{-2}$	$1.60360094 \times 10^{-2}$	$1.55836165 \times 10^{-1}$	$1.07496241 \times 10^{-2}$
	$1.03818923 \times 10^{-1}$	$-5.22193797 \times 10^{-2}$	$-1.11454939 \times 10^{-2}$	$-1.32430941 \times 10^{-2}$
	$-1.79913528 \times 10^{-2}$	$5.61168343 \times 10^{-2}$	$2.60032147 \times 10^{-1}$	$-7.01491013 \times 10^{-2}$
	$-1.26845524 \times 10^{-1}$	$2.05462247 \times 10^{-1}$	$-7.60471150 \times 10^{-2}$	$3.74383740 \times 10^{-2}$
	$-2.99654901 \times 10^{-1}$	$1.09292649 \times 10^{-1}$	$-2.23372858 \times 10^{-2}$	$1.03836328 \times 10^{-1}$
	$2.15664014 \times 10^{-1}$	$2.84033138 \times 10^{-2}$	$7.57873580 \times 10^{-2}$	$-2.20980458 \times 10^{-2}$
	$-5.43334223 \times 10^{-2}$	$-2.74950489 \times 10^{-2}$	$-3.19890827 \times 10^{-1}$	$5.89186065 \times 10^{-2}$
	$-9.21700075 \times 10^{-2}$	$-1.05074167 \times 10^{-1}$	$1.68249488 \times 10^{-3}$	$1.04820758 \times 10^{-1}$
	$-6.24499172 \times 10^{-2}$	$-1.86762381 \times 10^{-2}$	$2.91833598 \times 10^{-2}$	$-5.47248544 \times 10^{-3}$
	$-5.12562282 \times 10^{-2}$	$9.93735064 \times 10^{-3}$	$8.50876123 \times 10^{-2}$	$-5.97740635 \times 10^{-2}$
	$2.58028973 \times 10^{-2}$	$-2.03926042 \times 10^{-2}$	$2.73750186 \times 10^{-1}$	$1.54974297 \times 10^{-1}$
	$6.13427088 \times 10^{-2}$	$-5.41452393 \times 10^{-2}$	$9.14488211 \times 10^{-2}$	$-3.80940177 \times 10^{-2}$]

Figure 4. Token for "detail" from text vectorization.

Figure 5 presents the evaluation metrics obtained. The overall accuracy, which represents the percentage of correctly classified paragraphs, was 86%. The precision, indicating the proportion of correct predictions among all predicted positives, averaged 91%, ranging from 85% for K_A to fully correct predictions for K_C. Recall ranged from 22% to 96%, with an overall F1 score averaging 0.7, more precisely 0.9 for managerial predictions, 0.81 for commercial, and 0.36 for technical aspects.

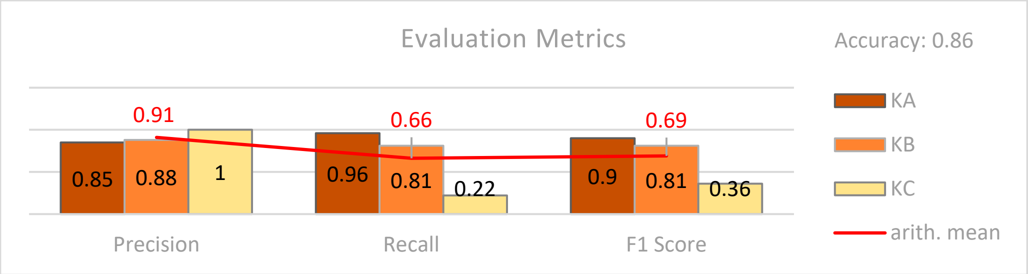


Figure 5. Evaluation metrics chart.

For selected sentences such as "The contractor shall compile in the BEP, for approval, the file formats that will be delivered to the client at each stage of the project," "Each model should be exported to the open Industry Foundation Classes (IFC) format," and "The final

BEP shall include information related to ‘Project Roles & Responsibilities’,” the model successfully classified them into the correct categories: K_A (managerial), K_B (commercial), and K_C (technical), as outlined by the ISO 19650 standard. The predicted classifications are shown in Table 1.

Table 1. Model-predicted categories for provided sentences.

No.	Sentence	Predicted Category
1	The contractor shall compile in the BEP, for approval, the file formats that will be delivered to the client at each stage of the project.	K_A
2	Each model should be exported to the open Industry Foundation Classes (IFC) format.	K_A
3	The final BEP shall include information related to ‘Project Roles & Responsibilities’.	K_A
4	All issues should be promptly reported to the client in the common data environment.	K_A

where K_A refers to managerial aspects, K_B to commercial aspects, and K_C to technical aspects.

Subsequently, the model was tasked with proposing sentences related to the provided phrases: “energy analysis,” “simulation of work sequences,” and “architectural model designed in Revit.” It searched the available database for similar sentences using cosine similarity, and if none were sufficiently similar, it generated new ones. The results, including predicted categories, are displayed in Table 2.

Table 2. Model-predicted sentences and categories for provided phrases.

No.	Phrase	Predicted Category	Most Similar Paragraph or New Paragraph	Similarity
1	Energy analysis	K_B	Energy analysis.	0.95
2	Simulation of work sequences	K_A	Project controls: The model will be capable of being utilized for identifying temporary as well as permanent works and provide a critical path analysis of site activities to prevent “trade clashes” and to provide the most efficient way of arranging working areas for operatives.	0.92
3	Architectural model designed in Revit	K_A	Developing constituent parts of the information model in connection with specific tasks.	0.91

where K_A refers to managerial aspects, K_B to commercial aspects, and K_C to technical aspects.

4. Discussion

The objective of this study was to propose an automatic text classification of Exchange Information Requirements (EIRs) according to ISO 19650 categories, using Support Vector Machines (SVMs) combined with text vectorization via Word2Vec. These methods proved sufficient to achieve this goal. In addition, the authors aimed to validate the model’s potential by examining its predictions. While the results may vary with a larger dataset, the findings from the current, smaller dataset (with 14,568 paragraphs identified) are promising. As shown in Figure 5, the model achieved an overall accuracy of 86%, indicating a significant percentage of correctly classified paragraphs. The model was particularly effective in identifying managerial aspects, where it achieved an F1 score of 0.9, which is the harmonic mean between precision and recall. However, the 100% precision for the technical category must be considered in conjunction with its 22% recall. This can indicate that the model infrequently predicts this category, but when it does, the predictions are correct.

Understanding the reasons behind these results requires an examination of the complexity of the data and the method’s limitations. The managerial category, which produced the best results, is typically the largest among K_A, K_B, and K_C. It is often written in language that is more easily interpreted by the Word2Vec algorithm and contain fewer tables (apart from the critical “roles and responsibilities” matrix) and less technical terminology than the other two categories. Both the commercial and technical categories may contain overlapping information, such as file formats, which can pertain both to data delivery schedules (K_B) and model requirements (K_C). Furthermore, the technical category, which yielded the poorest results in the evaluation metrics, is usually the shortest, contains specialized terminology, and often consists of tables. These tables, which present the levels of detail for geometric and alphanumeric information for various model components, are less conducive to classification through cosine similarity.

To improve performance, future iterations of the model could focus on balancing the dataset by increasing the pool of data from the technical category. Additionally, key terms or phrases associated with each category could be explicitly highlighted during training. For example, phrases such as "roles and responsibilities" could be tied to K_A, "BIM objectives" to K_B, and "file formats" to K_C.

When analyzing the model's predictions for the provided example sentences, as presented in Table 1, in sentence 1, the phrase "file formats" was used to test if the model would classify it under K_C (technical), yet it correctly classified it under K_A, recognizing the managerial context. Sentence 2, which involved file formats, should have been classified as technical, but the model assigned it to K_A, indicating an area for improvement. The third and fourth sentences were correctly classified under the category K_A. The authors conclude that three out of four sentences were acceptably classified.

Han et al. examined the application of Word2Vec and, while highlighting its efficiency in capturing semantic information, also pointed out a limitation: its inability to account for word order in sentences [24]. They observed that Word2Vec performs well with short sentences but struggles with more complex ones. In future studies, where the model will be trained on larger datasets, conducting a sensitivity analysis would be valuable to assess the impact of changes in input parameters on the model's performance, including sentence length and complexity, particularly with regard to technical terminology. It would also be beneficial to examine the effects of other parameters on classification, such as dataset size, the adjustment or the r parameter in SVM (which controls the margin between classes), the size of labelled data within each category, and the quality of the labelled data.

Table 2 presents the suggested similar paragraphs for the phrases provided. In the first case, the model identified "energy analysis" as a BIM use, which referred to specific requirements outlining how BIM models should be utilized for project management. These typically consist of a list of BIM uses, such as energy analysis, clash detection, and acoustic analysis, along with brief definitions of each. This section of the EIR pertains to commercial aspects; therefore, the phrase was correctly classified. However, the slight deviation in cosine similarity (0.95), despite an identical match, could be due to other paragraphs with similar semantics, affecting the similarity score. Sentence 2 returned an acceptable match regarding work sequences, though it did not fully capture the intended meaning of "simulation," indicating an area for refinement. Sentence 3 exhibited the model's limitation, as the proposed paragraph did not align with the query's intent, underscoring the importance of domain-specific training data.

This study acknowledges these limitations and identifies areas for improvement in future iterations, particularly in the domain of text generation. Future research will incorporate the use of GPT for more advanced text generation, which is expected to address some of the shortcomings encountered in this study.

5. Conclusions

This study investigates the application of Word2Vec for text vectorisation, Support Vector Machines (SVMs) for text classification, and the use of Word2Vec with cosine similarity for text generation. The findings demonstrate how Artificial Intelligence (AI) tools and methods can enhance information management in construction projects, specifically through the use of advanced methodologies such as Building Information Modelling (BIM) and Digital Twins.

The proposed model, which supports the creation of Exchange Information Requirements (EIR), particularly by categorizing text into managerial, commercial, and technical aspects according to the ISO 19650 standard, shows considerable potential. Even with a relatively small training dataset, the classification achieved an average F1 score—a harmonic mean of precision and recall—of 0.7, which is a promising result. Although this represents a limited sample, the initial tests indicate that the model is capable of classification, making it a valuable stepping stone for further research. This study highlights the effectiveness of Word2Vec for text vectorization and SVM with an RBF kernel for classification, which demonstrates the potential of these methods. However, the authors recognize the need to expand this exploration by integrating more advanced AI models, such as BERT, which could provide enhanced semantic understanding and overall classification accuracy. Additionally, it would be worthwhile in future research to investigate the impact of varying input data on the model's performance.

This study contributes both theoretically and practically to the field of information management in construction. Theoretically, it extends the application of AI in automating Exchange Information Requirements drafting for BIM/Digital Twin projects. Practically, this implementation could facilitate the adoption of these methodologies in the AECO industry, addressing existing barriers such as limited knowledge, the shortage of specialists, and alignment with contemporary trends, such as the European Union's digital transition.

Despite its contributions, this study is not without limitations. The relatively small dataset used for training constrains the generalizability of the results, and the current model requires further refinement to effectively handle larger and more diverse datasets. Additionally, while cosine similarity calculated using Word2Vec indicates semantic relationships between vectorized phrases, the generated sentences are not yet sufficiently accurate to be used directly in EIR documents. This points to the need for improvements in the text generation process, where more sophisticated models, such as GPT, could be employed to refine the generation of EIR fragments and improve overall text quality.

Looking forward, future research should focus on several key areas. First, the use of larger, more comprehensive datasets will be essential for improving the robustness and generalizability of the model. Secondly, exploring more advanced AI models, such as BERT and GPT, could provide deeper semantic understanding and enhance both the classification and generation of text. Our own future work will focus on integrating these advanced models into the automated EIR creation process. Furthermore, incorporating new EIRs into the database will be crucial for ensuring that the AI tools can adapt to changing trends in BIM/Digital Twin approaches.

In summary, while this study presents a robust method, there remains significant potential for future research to build on its findings. The implementation of such tools could significantly enhance the efficiency of complex construction processes involving advanced methodologies like BIM.

Author Contributions: Conceptualization, E.M.-K.; Methodology, E.M.-K.; Software, E.M.-K.; Validation, E.M.-K.; Formal Analysis, E.M.-K.; Investigation, E.M.-K.; Resources, E.M.-K.; Data Curation, E.M.-K.; Writing—Original Draft Preparation, E.M.-K.; Writing—Review and Editing, K.Z.; Visualization, E.M.-K.; Supervision, K.Z.; Project Administration, E.M.-K. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The original contributions presented in this study are included in the article; further inquiries can be directed to the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Renda, A.; Schwaag Serger, S.; Tataj, D.; Morlet, A.; Isaksson, D.; Martins, F.; Mir Roca, M.; Hidalgo, C.; Huang, A.; Dixson-Declève, S.; et al. *Industry 5.0, a Transformative Vision for Europe—Governing Systemic Transformations towards a Sustainable Industry*; European Commission, Directorate-General for Research and Innovation: Brussels, Belgium, 2021. [CrossRef]
2. Mitera-Kiełbasa, E.; Zima, K. BIM Policy Trends in Europe: Insights from a Multi-Stage Analysis. *Appl. Sci.* **2024**, *14*, 4363. [CrossRef]

3. *ISO 19650-1:2018*; Organization and Digitization of Information about Buildings and Civil Engineering Works, Including BIM—Information Management Using Building Information Modelling—Part 1: Concepts and Principles. International Organization for Standardization: Geneva, Switzerland, 2018.
4. Moreno, C.; Olbina, S.; Issa, R.R. BIM Use by Architecture, Engineering, and Construction (AEC) Industry in Educational Facility Projects. *Adv. Civ. Eng.* **2019**, *2019*, 1392684. [CrossRef]
5. Suganya, R.; Buhari, S.M.; Rajaram, S. Different Applications and Importance of Digital Twin. In *Digital Twin Technology*; Wiley: Hoboken, NJ, USA, 2022; pp. 189–203. [CrossRef]
6. Opoku, D.G.J.; Perera, S.; Osei-Kyei, R.; Rashidi, M.; Famakinwa, T.; Bamdad, K. Drivers for Digital Twin Adoption in the Construction Industry: A Systematic Literature Review. *Buildings* **2022**, *12*, 113. [CrossRef]
7. Borkowski, A.S. Evolution of BIM: Epistemology, genesis and division into periods. *J. Inf. Technol. Constr.* **2023**, *28*, 646–661. [CrossRef]
8. Soon Ern, P.A.; Ooi, Y.Y.; Al-Ashmori, Y.Y. Comparative Study on the Perspective towards the Benefits and Hindrances of Implementing Building Information Modelling (BIM). *J. Sustain. Constr. Eng. Technol.* **2020**, *11*, 194–205.
9. Biswas, H.K.; Sim, T.Y.; Lau, S.L. Impact of Building Information Modelling and Advanced Technologies in the AEC Industry: A Contemporary Review and Future Directions. *J. Build. Eng.* **2024**, *82*, 108165. [CrossRef]
10. *ISO 29481-1:2016*; Building Information Models—Information Delivery Manual—Part 1: Methodology and Format. International Standard Organization: Geneva, Switzerland, 2016.
11. Gohel, P.; Dabral, R.; Lad, V.H.; Patel, K.A.; Patel, D.A. A comprehensive review on application of artificial intelligence in construction management using a science mapping approach. In *Artificial Intelligence Applications for Sustainable Construction*; Elsevier: Amsterdam, The Netherlands, 2024; pp. 285–300. [CrossRef]
12. Rangasamy, V.; Yang, J.B. The convergence of BIM, AI and IoT: Reshaping the future of prefabricated construction. *J. Build. Eng.* **2024**, *84*, 108606. [CrossRef]
13. Ma, S.; Wang, X.; Wang, X.; Liu, H.; Zhang, R. A Framework for Diagnosing Urban Rail Train Turn-Back Faults Based on Rules and Algorithms. *Appl. Sci.* **2021**, *11*, 3347. [CrossRef]
14. Zheng, Z.; Lu, X.Z.; Chen, K.Y.; Zhou, Y.C.; Lin, J.R. Pretrained domain-specific language model for natural language processing tasks in the AEC domain. *Comput. Ind.* **2022**, *142*, 103733. [CrossRef]
15. Dolhopolov, S.; Honcharenko, T.; Terentyev, O.; Savenko, V.; Rosynskyi, A.; Bodnar, N.; Alzidi, E. Multi-Stage Classification of Construction Site Modeling Objects Using Artificial Intelligence Based on BIM Technology. In *Proceedings of the 2024 35th Conference of Open Innovations Association (FRUCT), Tampere, Finland, 24–26 April 2024*; IEEE: Piscataway, NJ, USA, 2024; pp. 179–185. [CrossRef]
16. Bartal, A.; Jagodnik, K.M.; Chan, S.J.; Dekel, S. AI and narrative embeddings detect PTSD following childbirth via birth stories. *Sci. Rep.* **2024**, *14*, 8336. [CrossRef]
17. Huang, A.H.; Wang, H.; Yang, Y. A Large Language Model for Extracting Information from Financial Text. *Contemp. Account. Res.* **2023**, *40*, 806–841. [CrossRef]
18. Piazzi, M.A.; Feng, H.; Kassem, M. An investigation of concepts for the specification of graphical exchange information requirements in building information modelling. *J. Inf. Technol. Constr.* **2022**, *27*, 662–684. [CrossRef]
19. Goonetillake, J.F.; Renb, G.; Lia, H.; Yaob, J. A prototype tool to embed digital exchange information requirements in construction projects. In Proceedings of the 30th EG-ICE: International Conference on Intelligent Computing in Engineering, London, UK, 4–7 July 2023; pp. 1–10.
20. Tomczak, A.; v Berlo, L.; Krijnen, T.; Borrmann, A.; Bolpagni, M. A review of methods to specify information requirements in digital construction projects. *IOP Conf. Ser. Earth Environ. Sci.* **2022**, *1101*, 092024. [CrossRef]
21. Luhn, H.P. The Automatic Creation of Literature Abstracts. *IBM J. Res. Dev.* **1958**, *2*, 159–165. [CrossRef]
22. Shakil, H.; Farooq, A.; Kalita, J. Abstractive Text Summarization: State of the Art, Challenges, and Improvements. *Neurocomputing* **2024**, *603*, 128255. [CrossRef]
23. Iqbal, T.; Qureshi, S. The survey: Text generation models in deep learning. *J. King Saud. Univ. Comput. Inf. Sci.* **2022**, *34*, 2515–2528. [CrossRef]
24. Han, M.; Zhang, X.; Yuan, X.; Jiang, J.; Yun, W.; Gao, C. A survey on the techniques, applications, and performance of short text semantic similarity. *Concurr. Comput.* **2021**, *33*, e5971. [CrossRef]
25. Wang, Z.; Dou, J.; Zhang, Y. Unsupervised Sentence Textual Similarity with Compositional Phrase Semantics. In Proceedings of the 29th International Conference on Computational Linguistics, Gyeongju, Republic of Korea, 12–17 October 2022; pp. 4976–4995.
26. Mikolov, T.; Chen, K.; Corrado, G.; Dean, J. Efficient Estimation of Word Representations in Vector Space. In Proceedings of the International Conference on Learning Representations, Scottsdale, AZ, USA, 2–4 May 2013. [CrossRef]
27. Mikolov, T.; Chen, K.; Corrado, G.; Dean, J. Word2vec Code. 2024. Available online: https://code.google.com/archive/p/word2vec/ (accessed on 24 May 2024).
28. Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G.; Dean, J. Distributed Representations of Words and Phrases and their Compositionality. *Adv. Neural Inf. Process. Syst.* **2013**. [CrossRef]
29. Zima, K.; Mitera-Kiełbasa, E. Employer's information requirements: A case study implementation of BIM on the example of selected construction projects in Poland. *Appl. Sci.* **2021**, *11*, 10587. [CrossRef]

30. Polish Association of Construction Engineers and Technicians; Polish Association of Construction Employers; Association of Polish Architects. BIM Standard PL. 2020. Available online: https://globalbim.org/info-collection/bim-standard-pl/ (accessed on 20 October 2024).
31. Zima, K.; Mitera-Kiełbasa, E. Level of Information Need for BIM Models: Australia, New Zealand and ISO 19650. *Civ. Environ. Eng. Rep.* **2022**, *32*, 1–3. [CrossRef]
32. Poole, D.L.; Mackworth, A.K. *Artificial Intelligence*; Cambridge University Press: Cambridge, UK, 2023. [CrossRef]
33. Vapnik, V.N. *The Nature of Statistical Learning Theory*; Springer: New York, NY, USA, 2000. [CrossRef]
34. Schölkopf, B.; Smola, A.J. *Learning with Kernels*; The MIT Press: Cambridge, MA, USA, 2018. [CrossRef]
35. Burges, C.J.C. A Tutorial on Support Vector Machines for Pattern Recognition. *Data Min. Knowl. Discov.* **1998**, *2*, 121–167. [CrossRef]
36. Cortes, C.; Vapnik, V. Support-vector networks. *Mach. Learn.* **1995**, *20*, 273–297. [CrossRef]
37. Scikit-Learn Developers (BSD License). Scikit-Learn. Machine Learning in Python. 2024. Available online: https://scikit-learn.org/ (accessed on 13 September 2024).
38. Hobson, L.; Howard, C.; Hapke, H. *Przetwarzanie Języka Naturalnego w Akcji. Rozumienie, Analiza i Generowanie Tekstu w Pythonie Na Przykładzie Języka Angielskiego*, 1st ed.; Wydawnictwo Naukowe PWN: Warsaw, Poland, 2021.

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.