NLP Approaches to align Construction Classification Systems

Stefan Fuchs, Dan Jones, Robert Amor

July 2021

1 Introduction

Classification systems are an essential part of the construction industry. They are used for managing project information, specifying building products, building information modelling and more. Since there is no common standard for classification systems, there are interoperability issues between the classification systems of different countries (e.g. UniClass in the UK, OmniClass in the US, and NZAMS in New Zealand). To compare or integrate projects or software using different classification systems, it becomes necessary to align the various classifications by mapping the classes to each other. This mapping can be a complicated task taking into account that the classifications can offer different views on the construction industry and all related aspects. The abstraction level and coverage can vary enormously. The 336 classes in NZAMS oppose 11,840 classes in UniClass and even more in OmniClass. NZAMS uses five upper-level classes: Site & Structure, Exterior, Interior, Services, and External Assets & Sundries. In comparison, UniClass offers a set of twelve tables including Elements & Functions (EF), Systems, Products, Tools & Equipment, and Activities. Accordingly, there are many-to-many relationships between the classes, and mappings exist between different levels of the classification systems. Various sources of information can help to identify valid alignments. Classification titles, definitions, and neighbouring class nodes (i.e. parent, siblings, and children) are information sources coming first to mind. The textual nature of these information types and the need to find semantically equivalent classes makes this problem a natural fit for Natural Language Processing (NLP) techniques.

This project explores prior research in the semantic alignment of classification systems, ontologies, database schemata and more. We introduce a semantic similarity-based mapping approach based on the recent successes of deep learning for understanding the semantics of text in natural language. While our system offers a significant improvement over traditional string-based comparisons, the difficulty of identifying a suitable threshold to decide if a mapping is valid or not suggests that human supervision is still necessary. We recommend refining the deep learning models on a large corpus of construction-related text to teach it the necessary domain-specific knowledge. Additionally, better class descriptions and auxiliary information such as existent mappings to other classification systems could improve the overall matching accuracy and minimize the required manual effort.

2 Background

Aligning these classification systems falls into the research areas of taxonomy, schema, ontology, and knowledge graph alignment, mapping, or matching. Taxonomies, schemata, ontologies, and knowledge graphs are different methods in computer science to formalise knowledge. While the classification systems are formally seen as taxonomies, the most active and closely related research area is ontology alignment. The Ontology Alignment Evaluation Initiative (OAEI) (Ontology Alignment Evaluation Initiative, 2021) produces ontology alignments in different disciplines every year to evaluate the state-of-the-art ontology alignment systems.

There is a wide range of different classifications of the existent approaches. Rahm & Bernstein (2001) categorised the mapping methods based on element vs structural and linguistic vs constraint-based features. Element level comparisons compare the nodes themselves by comparing the labels, descriptions, and more. In contrast, the structural level refers to relations with neighbouring nodes, the hierarchy level, and more. Some approaches use linguistic information to perform string-based comparisons of labels and descriptions using techniques like stemming and tokenisation. Others calculate the similarity of vectorised text using pre-trained word embeddings. Auxiliary information such as thesauri or dictionaries is also common in this category. Constraint-based comparisons use information like data types, cardinality, and hierarchy levels. Other common approaches include extensional-based (i.e. comparing instances), graph-based (i.e. including weighted neighbourhood information), rule-based (e.g. first-order logic rules, regular expressions), mapping repair strategies (e.g. logic reasoner), user interactions, and sequential matching (i.e. high confidence matches as anchors) (Bernstein et al., 2011).

3 Methodology

The Building Innovation Partnership (BIP) investigates the use of classification systems for managing built assets in New Zealand. A tool was developed to allow the customers to define their asset information requirements (AIR tool). While this tool utilises the NZAMS classification system, designed by Masterspec, a survey identified OmniClass and UniClass as the most commonly used classification systems within NZ. Accordingly, mapping between various classification systems is required to allow NZ wide application of the AIR tool and give organisations the freedom to change the way their assets are classified. Our initial research suggests no mappings between NZAMS and OmniClass or UniClass are available. So, a manual process was started to map the NZAMS and

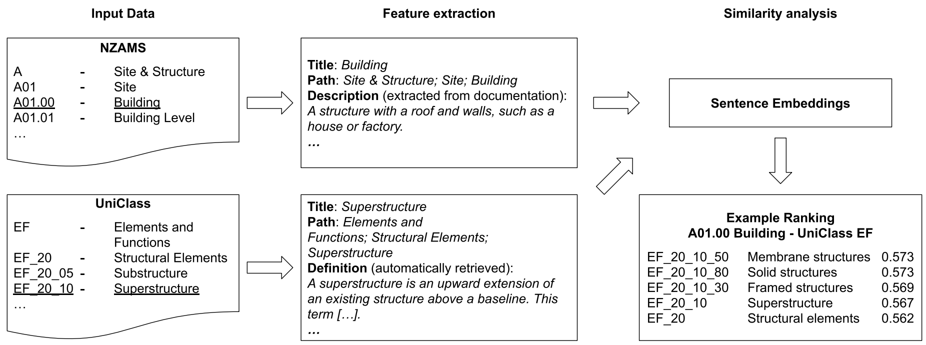

FIGURE 1: Mapping process example

UniClass classification systems. Since this process was identified as too time-consuming, we are researching methods to automate this process using those manually identified mappings as ground truth data.

Although there are many ontology mapping systems available (see Bernstein et al. (2011); Ardjani et al. (2015); Angermann & Ramzan (2017); Khoudja et al. (2019); Mohammadi & Rezaei (2020) for extensive comparisons and summaries), most systems are too restrictive for our use case. This originates mainly from the common limitation to 1:1 mappings. Our use case additionally requires n:m mappings, including the assignment to subclasses whenever no equivalent classes can be identified.

First, we tested different traditional and ML matching systems. As the tested traditional system was optimised for precision, only a few matches were found. In contrast, refining ML-based systems was problematic since we have only a small number of mappings in our ground truth. So, the ML models overfitted the training data. As an alternative, we utilised recent developments on BERT- and RoBERTa-based universal sentence embeddings for semantic comparisons of the classification titles. The semantic similarity between text labels can be computed directly using various pre-trained models. We then included auxiliary information from the classification documentation of NZAMS and automatically retrieved definitions for UniClass categories since UniClass descriptions are only sparsely available. Finally, we included structural information in the form of class titles for parents, siblings and children. The final mappings were identified using a grid search over different threshold values and feature weights. Figure 1 gives an overview of the mapping process using an example mapping.

3.1 Gold standard

NZAMS and UniClass were selected as the test case of the study. NZAMS will be the reference classification. The NZAMS classes are mapped against each table in UniClass. The mapping between NZAMS and the UniClass - Elements and function (EF) table is used as the gold standard. Two researchers selected valid mappings based on similarities calculated in Microsoft Excel. Figure 2 shows an excerpt of the resulting ground truth table. These mappings were used in the subsequent experiments to calculate the F2-Measure. We chose a beta value of 2 for the F-Measure to weigh recall higher than precision. Since we consider a review and post-processing of the identified mappings, this setting helps to prevent missing mappings.

3.2 Existing approaches

To test existing approaches, we were looking for current state-of-the-art ontology alignment systems with a focus on high recall. Another requirement was that the tool is freely available, open-source, and described in a research paper. Based on our hypothesis, the use of sentence embeddings is another plus point. First, we selected VeeAlign (Iyer et al., 2020), a system with state-of-the-art results in the OAEI 2020 conference track (Ontology Alignment Evaluation Initiative, 2020). Furthermore, LogMap-ML (Chen et al., 2021) was chosen since it significantly improves the recall of LogMap (Jiménez-Ruiz & Cuenca Grau, 2011), a traditional matching system. Furthermore, this choice allows us to test LogMap simultaneously.

A conversion of the classification systems into an ontology format was required to use the introduced tools for our use case. The selected alignment systems use OWL as the input format. Accordingly, the python library owlready2 was used to create the ontology from the classification systems. The classification titles were used as class names, and the hierarchy was modelled as subclass relationships.

LogMap (Jiménez-Ruiz & Cuenca Grau, 2011)

LogMap is a rule-based alignment system using word similarity and a repair mechanism based on logic inferences. The mappings are produced in a sequential process. First, an over-estimation is computed based on lexical similarity and synonyms. High-confidence anchor mappings are extracted from the over-estimation and used as the foundation for further mappings. Repair steps are applied on the anchor mappings as well as the final mappings.
Results: Mappings between NZAMS and the Uniclass Elements and Functions table were produced with LogMap 4.0. The following results indicate that strict, traditional, string-based comparisons are not suitable for our fuzzy mapping objectives.

- **10 over-estimations**: Stair - Stairs, Substructure - Substructure, Ramp - Ramps, Chimney - Chimneys, Barrier - Barriers, Light - Lighting, Floor - Floors, Sign - Signage, Interior Walls - Walls, Paving - Pavements
- **3 anchors**: Stair - Stairs, Substructure - Substructure, Ramp - Ramps
- **4 additional mappings**: Chimney - Chimneys, Barrier - Barriers, Light - Lighting, Floor - Floors

Number	Description	Code	Title	Similarity	Match?
A -	Site & Structure	EF_20_05	Substructure	0.8052	
A -	Site & Structure	EF_20	Structural elements	0.6175	Y
A -	Site & Structure	EF_20_50	Bridge structures	0.5074	
A -	Site & Structure	EF_20_10_15	Composite structures	0.4597	
A -	Site & Structure	EF_20_10_30	Framed structures	0.4597	
A -	Site & Structure	EF_20_10_50	Membrane structures	0.4597	
A -	Site & Structure	EF_20_10_75	Shell structures	0.4597	
A -	Site & Structure	EF_20_10_80	Solid structures	0.4597	
A01	Site			0.0000	
A01.00	Building	EF_20_10	Superstructure	0.0000	Y
A01.01	Building Level	EF_30_20	Floors	0.0000	Y
A01.02	Room			0.0000	
A01.03	Outdoor Precinct			0.0000	
A01.04	Site Preparation			0.0000	
A01.04.01	Ground retainment temporary	EF_50_30	Above-ground drainage collection	0.2593	
A01.04.01	Ground retainment temporary	EF_50_35	Below-ground drainage collection	0.2593	
A01.04.02	Underpinning			0.0000	
A01.04.03	Shoring			0.0000	
A01.04.04	Diversion			0.0000	
A01.04.05	Dewatering			0.0000	
A01.04.06	Site Clearance			0.0000	
A01.04.07	Excavation			0.0000	
A01.04.08	Bulk fill			0.0000	
A01.04.09	Demolition			0.0000	
A02	Substructure	EF_20_05	Substructure	1.0000	Y
A02	Substructure	EF_20_10	Superstructure	0.9098	
A02	Substructure	EF_20_50	Bridge structures	0.7501	
A02	Substructure	EF_20_10_15	Composite structures	0.7408	
A02	Substructure	EF_20_10_30	Framed structures	0.7408	
A02	Substructure	EF_20_10_50	Membrane structures	0.7408	
A02	Substructure	EF_20_10_75	Shell structures	0.7408	
A02	Substructure	EF_20_10_80	Solid structures	0.7408	
A02.01	Piling	EF_20_05_30	Foundations	0.7621	Y
A02.01	Piling	EF_30_60_95	Vehicular paving	0.7621	
A02.01	Piling	EF_60_30	Rail and paving heating	0.7377	
A02.01	Piling	EF_30	Roofs, floor and paving elements	0.7266	

Figure 2: Ground truth mappings. Number and Description refer to the NZAMS classification and Code and Title to UniClass - EF. The yellow highlighted rows are valid mappings, non-highlighted rows were not selected as mappings, and rows with empty codes and titles indicate no similar terms in the UniClass tables were identified.

- **3 discarded mappings**: Sign - Signage, Interior Walls - Walls, Paving - Pavements
- **7 final mappings**: Stair - Stairs, Substructure - Substructure, Ramp - Ramps, Chimney - Chimneys, Barrier - Barriers, Light - Lighting, Floor - Floors

LogMap-ML (Chen et al., 2021)

LogMap-ML is a machine learning-based extension of LogMap. GloVe (Pennington et al., 2014) or OWL2Vec (Chen et al., 2020) embeddings can be used to represent the ontology text labels and paths. These vectors were then fed into several neural network architectures trained with the LogMap anchor classes and randomly generated negative samples. Then, the model was used to predict similarity scores for the candidate mappings (i.e. LogMap over-estimation).
Results: Due to the lack of anchor mappings extracted by the LogMap system (i.e. Subsection 3.2), the ground truth was used to train the neural network. There are 525 valid mappings between NZAMS and the UniClass - EF in the ground truth. Additionally, 1203 negative samples were generated, resulting in 1728 training samples. The following results were achieved using the LogMap-ML standard configuration:

- **MLP**: threshold: 0.42, precision: 0.560, recall: 0.953, f1: 0.705, acc: 0.602
- **BiRNN**: threshold: 0.76, precision: 0.961, recall: 1.000, f1: 0.980, acc: 0.980
- **AttBiRNN**: threshold: 0.78, precision: 0.910, recall: 1.000, f1: 0.953, acc: 0.951
- **SiameseMLP**: threshold: 0.54, precision: 0.955, recall: 0.977, f1: 0.966, acc: 0.965
- **SiameseAttBiRNN**: threshold: 0.72, precision: 0.942, recall: 0.942, f1: 0.942, acc: 0.942

While the mappings with the trained table are relatively good, Table 1 shows the overfitting problem of this system originating from little training data. It compares the NZAMS class 'Site and Structure' mapped to the trained table UniClass - EF and the unseen table UniClass - Systems. Terminologically related classes from the UniClass - EF have extremely high values, and unrelated terms are close to zero. In contrast, all similarity scores with the unseen UniClass - Systems classes are very low.

VeeAlign (Iyer et al., 2020)

VeeAlign is a supervised learning approach to mapping ontologies. They use the Universal Sentence Encoder (Cer et al., 2018) to generate the input for a siamese network. This network uses node and path attention layers for context feature

Similarity with NZAMS class - 'Site and Structure'			
UniClass - Elements and functions	**Score**	**UniClass - Systems**	**Score**
Substructure	0.996	Structural systems	0.011
Superstructure	**1.000**	Groundworks and earthworks systems	0.006
Structural elements	**0.997**	Substructure systems	0.007
Membrane structures	0.996	Earthworks excavating systems	0.006
Composite structures	0.910	Temporary structural systems	0.002
Framed structures	0.980	Structure covering and finishing systems	**0.023**
Solid structures	0.981	External signage and interpretation systems	0.002
Equipment	0.000	Stair and ramp structure systems	**0.178**
Signage	0.000	Earthworks and remediation and temporary systems	0.002
Shell structures	0.850	Internal architectural signage systems	0.001

Table 1: LogMap-ML similarity scores between NZAMS class - 'Site and Structure' and UniClass - EF and UniClass - Systems.

generation. A feedforward neural network is used for dimensionality reduction before calculating the final similarity scores. An experimentally determined threshold is used to make the final predictions and calculate the mean squared loss for the training accordingly.

Results: The pre-trained version of VeeAlign (OAEI-2020 - Conference track) was used to predict mappings between NZAMS and UniClass - EF. With a threshold of 0.59, an F2-Score of 21.2% was achieved. Furthermore, Table 2 shows example mappings between NZAMS and all UniClass tables with thresholds of 0.8 and 0.9 for qualitative evaluation. While the mappings are mostly reasonable, some errors can be detected (e.g. System photovoltaic - Hatch systems). Those errors increased drastically once a lower threshold was used to detect more true positives.

3.3 Semantic similarity-based alignment

Finally, our semantic similarity-based alignment yielded the most promising results of the conducted experiments. This approach utilises the recent successes of transformer-based pretrained language models like BERT (Devlin et al., 2018) and RoBERTa (Liu et al., 2019) and universal sentence embedding strategies like FlowBERT (Li et al., 2020), InferSent (Conneau et al., 2017), Universal Sentence Encoder (Cer et al., 2018), SBERT (Reimers & Gurevych, 2020), and SimCSE (Gao et al., 2021). Instead of training a classifier, we determine matches based on a threshold for the similarity between two classes (i.e. similar to VeeAlign). The best threshold for each experiment was determined by automatically searching the range from 0.4 to 0.6 with a step of 0.02. Table 3 shows the results of the experiments using the two state-of-the-art models, SBERT and SimCSE, and other reference models provided by the SentenceTransformer library. The embedding models were used to encode the class titles, and the cosine similarity was calculated. The top 10 matches with similarity scores above the threshold were compared to the gold standard to calculate the F2-Measure. These experiments indicated the suitability of the supervised SimCSE model for our task. The contrastive training objective of this model

Threshold 0.9		Threshold 0.8	
NZAMS	**UniClass**	**NZAMS**	**UniClass**
Equipment kitchen	Domestic cooking equipment	Fence Gates	Gate systems
Gutter Eaves	Eaves gutter brackets	Insulation wall lining	Cavity wall insulation systems
Insulation wall lining	Wall insulation systems	Insulation wall lining	Stainless steel insulating sandwich panels
Equipment laboratory	Medical and laboratory equipment	Baffle ceiling	Ventilated ceilings
Soffit	Soffit grilles	Warning device fire	Visual alarm signal devices
Traffic control system	Traffic management control systems	Manual call system	Emergency assistance call systems
Handrail	Handrails	Space wall	Walls
Clock system	Clock systems	Strapping ceiling	Wall safes
Irrigation system	Irrigation systems	Exterior wall	External wall grilles
Parking Meters	Parking meters	Gantry crane	Crane systems
Refrigeration system	Refrigeration systems	Roof structure	Roof hatches
Water meter	Water meters	Access Control	Access control units
Hatch wall	Wall hatches	Pneumatic system	Pneumatic waste collection systems
Fume extraction system	Fume extract systems	System photovoltaic	Hatch systems
Appliance sanitary	Sanitary appliance systems	Fume extraction system	Industrial fume extract systems
Plant sewage treatment	Sewage treatment plant	Wall cladding	Wall cladding systems
Pump drainage	Drainage pumps	Kerb	Light-duty kerb systems
Air handling unit	Air handling units	Earthing system	Retaining wall systems
Stairs and Balustrades	Stairs and ramps	Roof raftered	Aluminium eaves gutters
Road Paving	Vehicular paving	Pump HVAC	Brine-to-water heat pumps
Stair	Stair stringers	Stairs and Balustrades	Stair and ramp systems
Escalator	Escalators	Detection system fire	Gas detection and alarm systems
Structural component	Structural elements	Stairs and Balustrades	Stone stair treads
Unit fan coil	Fan coil units	Fume extraction system	Wastewater decanting systems
Roof access hatch	Roof hatches	Protective system	Protection systems
Trees	Trees	Wall cladding	Plasterboard panels
Mirrors	Mirrors	Windows and Exterior Doors	Door hardware systems
Emergency lighting	Emergency luminaires	Fire system	Fire bucket systems
Windows and Exterior Doors	Doors and windows	Protective system	Flotation systems
Substructure	Substructure	Shelving	Mobile aisle shelving
Shelving	Shelves	Roof raftered	Roof hatch systems
Door strongroom	Door formers	Fitting gas	Gas grills
Stairs and Balustrades	Stairs	Stairs and Balustrades	Escalators
Barrier	Barriers	Insulation wall lining	Wall lining systems
Air handling unit	Supply air handling units	Fire Hydrant	Above-ground fire hydrants
Structural member	Structural members	...	...
38 mappings		**742 mappings**	

Table 2: VeeAlign mappings between NZAMS and all UniClass tables

Sentence embedding type	Architecture	Threshold	F2-Score
SimCSE	princeton-nlp/sup-simcse-roberta-large	0.58	0.431
SentenceTransformer	msmarco-distilbert-base-v2	0.40	0.389
SentenceTransformer	stsb-mpnet-base-v2	0.50	0.378
SentenceTransformer	stsb-roberta-base-v2	0.46	0.374
SentenceTransformer	nli-mpnet-base-v2	0.52	0.373
SentenceTransformer	nli-roberta-base-v2	0.56	0.367
SentenceTransformer	nli-distilroberta-base-v2	0.56	0.343
SimCSE	princeton-nlp/unsup-simcse-roberta-large	0.40	0.319
GloVe	average_word_embeddings_glove.840B.300d	0.42	0.311
Multilingual Universal Sentence Encoder	distiluse-base-multilingual-cased-v2	0.54	0.294
SentenceTransformer	stsb-xlm-r-multilingual	0.58	0.294
SentenceTransformer	paraphrase-xlm-r-multilingual-v1	0.50	0.290
SentenceTransformer	paraphrase-distilroberta-base-v1	0.40	0.287
Pooling	nli-bert-large-cls-pooling	0.58	0.251
Pooling	nli-bert-large-max-pooling	0.58	0.249

Table 3: Comparison of different sentence embedding strategies, training data sets, model architectures, pooling strategies for mapping NZAMS to UniClass - Elements and functions. See https://www.sbert.net/docs/pretrained_models.html for more information

solves the anisotropy problem of BERT-based models. That means the embeddings are uniformly distributed in the vector space, allowing a more nuanced differentiation between similarities leading to better performance with a higher threshold (i.e. less false positives). As a next step, we added structural and auxiliary information to provide additional context information. As structural features, we concatenate the class title of siblings, children, the direct parent, and the full path. We considered term definitions as auxiliary information for an improved understanding of specialised domain terminology. For NZAMS, there are definitions available in the NZAMS documentation. By concatenating the NZAMS definition vector with the NZAMS title vector and comparing them with the UniClass title vector, an improvement of 4% F2-Score could be achieved.

In contrast, we could not identify a good source for UniClass class definitions. While some definitions are available on https://toolkit.thenbs.com/definitions, this web page does not contain many definitions for the Element and functions table. Accordingly, we compared WordNet and Wikipedia definitions with definitions retrieved from Google and ChatNoir. For WordNet and Wikipedia, we used the python libraries 'PyDictionary' and 'wikipedia'. Since PyDictionary does not allow phrases, we used this strategy either for titles consisting of one word only or split the title into separate words and retrieved the definitions for those words. The Wikipedia library offers summary and search functions. We used either the summary function directly for each class title or searched for a corresponding Wikipedia page first and retrieved the summary subsequently. We used the SimCSE model to retrieve the WordNet definitions and Wikipedia page suggestions that are most similar to the keywords 'construction', 'building', and 'engineering'. The Google definitions were retrieved using Google's featured snippet definitions or the snippet of the first search result. Figure 3 shows an example query contextualised by concatenating "define the term", the

Retrieval strategy	Threshold	F2-Score
NZAMS definition - UniClass title	0.52	0.4262
Google defintion	0.50	0.3543
Wiki summary OR Wiki search OR Similar WordNet meaning	0.46	0.3331
Wiki summary OR Similar WordNet meaning	0.46	0.3324
Wiki summary OR Wiki search	0.46	0.3314
Similar WordNet meaning for title words	0.44	0.3309
All WordNet synonyms for title words	0.46	0.3249
All WordNet meanings for title words	0.44	0.3150
Similar WordNet meaning for titles with one word OR Wiki search	0.46	0.2695
Wiki search	0.46	0.2614
Wiki search with fuzzy word match	0.48	0.2579
ChatNoir - Title, Top1	0.40	0.2378
ChatNoir - Title	0.42	0.2354
ChatNoir - Title + 'Definition'	0.42	0.2255
ChatNoir - Title, CW only	0.42	0.2204
ChatNoir - Title + 'Definition' + 'Building', CW only	0.48	0.2005
ChatNoir - Title + 'Definition' + 'Building'	0.44	0.1794
ChatNoir - Title + Parent title	0.50	0.1602

Table 4: Comparison of definition retrieval strategies. Evaluation on NZAMS - UniClass EF with Definition - Definition similarity. ChatNoir defaults: Concatenate Top2 results and search in ClueWeb12 (CW) and CommonCrawl 11/2015 (CC) indices (see https://www.chatnoir.eu/doc/api/ for additional information).

class title, and "in construction". As an alternative to the limited free Google APIs, we used the ChatNoir API Bevendorff et al. (2018). ChatNoir is an elastic search engine for web crawl corpora like ClueWeb12 and CommonCrawl. Table 4 shows the result of these comparisons. We found that the compound terms and ambiguous meanings of many class titles lead to relatively noisy definitions. Although the Google definitions could be contextualised and achieved subjectively and objectively better results, they contributed negatively to the end result compared to the NZAMS Definition - UniClass Title comparison. Since the identified free APIs for Google have low request limits, we opted for the Wikipedia and WordNet strategies for further experiments and the WordNet only strategy for the final mapping generation. This decision was made as no Wikipedia articles were found for most of the classes in the other UniClass tables. While the ChatNoir strategy was not followed up further since a simple contextualisation was not possible, more sophisticated retrieval strategies could help to retrieve better results from this source.

Finally, the cosine similarity score was calculated as a weighted sum over the similarities of the following seven features: 1) Title, 2) NZAMS definition and UniClass title, 3) Definition, 4) Path, 5) Parent, 6) Siblings, and 7) Children. We performed a grid search over all possible weight distributions with a step of 0.05, resulting in 230230 weight combinations times eleven threshold options using the supervised SimCSE model as the encoder. Table 5 shows the top 10 weight distributions indicating the importance of the definitions and paths as features. When the definition quality was decreased due to retrieving them automatically from WordNet and Wikipedia, the weights for the "definition - definition" similarity lost importance (see Table 6).

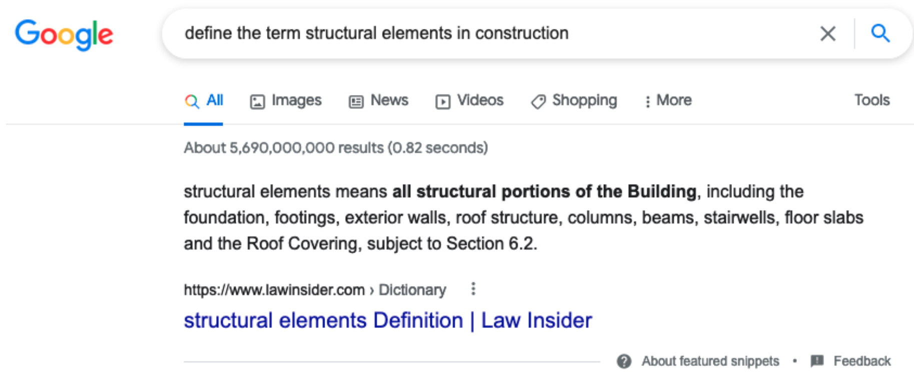

FIGURE 3: Example query to extract construction related definitions from featured snippets.

Title	**Def. - Title**	**Def.**	**Path**	**Parent**	**Siblings**	**Children**	**Threshold**	**F2-Score**
0.10	0.35	0.25	0.25	0.00	0.00	0.05	0.54	0.5215
0.10	0.35	0.20	0.30	0.00	0.00	0.05	0.54	0.5182
0.10	0.30	0.25	0.30	0.00	0.00	0.05	0.54	0.5155
0.05	0.50	0.15	0.25	0.00	0.00	0.05	0.54	0.5123
0.05	0.45	0.20	0.25	0.00	0.00	0.05	0.54	0.5113
0.05	0.40	0.20	0.30	0.00	0.00	0.05	0.54	0.5107
0.05	0.30	0.20	0.35	0.10	0.00	0.00	0.50	0.5045
0.00	0.40	0.20	0.30	0.05	0.00	0.05	0.52	0.5033
0.00	0.35	0.25	0.35	0.00	0.00	0.05	0.54	0.5006
0.00	0.30	0.30	0.35	0.00	0.00	0.05	0.54	0.4989

TABLE 5: Top 10 configurations for feature weights and thresholds with Google definitions for UniClass.

Title	**Def. - Title**	**Def.**	**Path**	**Parent**	**Siblings**	**Children**	**Threshold**	**F2-Score**
0.20	0.45	0.15	0.20	0.00	0.00	0.00	0.52	0.5137
0.15	0.35	0.20	0.30	0.00	0.00	0.00	0.52	0.5125
0.10	0.40	0.20	0.25	0.00	0.05	0.00	0.52	0.5093
0.10	0.35	0.25	0.25	0.00	0.05	0.00	0.52	0.5091
0.10	0.35	0.20	0.35	0.00	0.00	0.00	0.52	0.5071
0.10	0.35	0.20	0.25	0.05	0.05	0.00	0.50	0.5056
0.10	0.30	0.25	0.35	0.00	0.00	0.00	0.52	0.5051
0.10	0.30	0.20	0.35	0.05	0.00	0.00	0.50	0.5046
0.10	0.25	0.25	0.35	0.05	0.00	0.00	0.50	0.5022
0.00	0.60	0.05	0.30	0.00	0.00	0.05	0.52	0.5012

TABLE 6: Top 10 configurations for feature weights and thresholds with WordNet meanings and Wikipedia summaries for UniClass.

4 Conclusion and Future Research Directions

We developed a semantic mapping approach for construction classification systems that incorporates pre-trained deep learning models with title, structure and auxiliary information. While we could identify vital semantic mappings, these are currently not exact enough to be used directly in the AIR tool. A manual review is still necessary, especially since the threshold is difficult to specify. In many cases, we either got too many mappings or no mappings at all. Furthermore, subclass mappings were in some cases preferred over equivalent mapping. This effect could be mitigated using a post-processing step.

We plan further experiments to reduce the manual effort and move towards a fully automated solution in future developments. The definition retrieval strategy should be refined, and other auxiliary information sources should be considered. For example, instance data of real-world projects where the classification systems are applied, as well as existing mappings between OmniClass and UniClass, could help to improve the mappings (Bernstein et al., 2011). In addition, refining the word embeddings on domain knowledge could improve the accuracy. The utilised SimCSE embedding are trained in a contrastive manner on Wikipedia text. Using construction-related text might help to obtain similarities based on a more suitable word meaning and serve as automatic term disambiguation. Another common strategy is to repair the mappings based on logic inferences. Instead, a rule-based post-processing step, which integrates hierarchy information and anchor mappings, might offer a simple way to improve the results. Finally, the structural information was directly used to form multiple features. Graph neural networks could be used as an alternative method to provide contextualised weighted neighbour information (Cao et al., 2020).

References

Angermann, H., & Ramzan, N. (2017). *Taxonomy Matching Using Background Knowledge.* doi: 10.1007/978-3-319-72209-2

Ardjani, F., Bouchiha, D., & Malki, M. (2015). Ontology-Alignment Techniques: Survey and Analysis. *International Journal of Modern Education and Computer Science*, *7*(11), 67–78. doi: 10.5815/ijmecs.2015.11.08

Bernstein, P. A., Madhavan, J., & Rahm, E. (2011). Generic schema matching, ten years later. *Proceedings of the VLDB Endowment*, *4*(11), 695–701. doi: 10.14778/3402707.3402710

Bevendorff, J., Stein, B., Hagen, M., & Potthast, M. (2018, March). Elastic ChatNoir: Search Engine for the ClueWeb and the Common Crawl. In L. Azzopardi, A. Hanbury, G. Pasi, & B. Piwowarski (Eds.), *Advances in information retrieval. 40th european conference on ir research (ecir 2018).* Berlin Heidelberg New York: Springer.

Cao, Y., Liu, Z., Li, C., Liu, Z., Li, J., & Chua, T. S. (2020). Multi-channel graph neural network for entity alignment. *ACL 2019 - 57th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference*, 1452–1461. doi: 10.18653/v1/p19-1140

Cer, D., Yang, Y., Kong, S. y., Hua, N., Limtiaco, N., St. John, R., … Kurzweil, R. (2018). Universal sentence encoder for English. *EMNLP 2018 - Conference on Empirical Methods in Natural Language Processing: System Demonstrations, Proceedings*, 169–174. doi: 10.18653/v1/d18-2029

Chen, J., Hu, P., Jimenez-Ruiz, E., Holter, O. M., Antonyrajah, D., & Horrocks, I. (2020). OWL2Vec*: Embedding of OWL Ontologies. Retrieved from http://arxiv.org/abs/2009.14654 doi: 10.1007/s10994-021-05997-6

Chen, J., Jimenez-Ruiz, E., Horrocks, I., Antonyrajah, D., Hadian, A., & Lee, J. (2021). Augmenting Ontology Alignment by Semantic Embedding and Distant Supervision. *European Semantic Web Conference*, 392–408.

Conneau, A., Kiela, D., Schwenk, H., Barrault, L., & Bordes, A. (2017). Supervised learning of universal sentence representations from natural language inference data. *EMNLP 2017 - Conference on Empirical Methods in Natural Language Processing, Proceedings*, 670–680. doi: 10.18653/v1/d17-1070

Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv*, 4171–4186.

Gao, T., Yao, X., & Chen, D. (2021). SimCSE: Simple Contrastive Learning of Sentence Embeddings. Retrieved from http://arxiv.org/abs/2104.08821

Iyer, V., Agarwal, A., & Kumar, H. (2020). VeeAlign: A supervised deep learning approach to ontology alignment. *CEUR Workshop Proceedings*, *2788*, 216–224.

Jiménez-Ruiz, E., & Cuenca Grau, B. (2011). LogMap: Logic-based and scalable ontology matching. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *7031 LNCS*(PART 1), 273–288. doi: 10.1007/978-3-642-25073-6{_}18

Khoudja, M. A., Fareh, M., & Bouarfa, H. (2019). Ontology Matching using Neural Networks: Survey and Analysis. *Proceedings of the 2018 International Conference on Applied Smart Systems, ICASS 2018*(November), 24–25. doi: 10.1109/ICASS.2018.8652049

Li, B., Zhou, H., He, J., Wang, M., Yang, Y., & Li, L. (2020). On the Sentence Embeddings from Pre-trained Language Models. , 9119–9130. doi: 10.18653/v1/2020.emnlp-main.733

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., ... Stoyanov, V. (2019). RoBERTa: A Robustly Optimized BERT Pretraining Approach. (1). Retrieved from http://arxiv.org/abs/1907.11692

Mohammadi, M., & Rezaei, J. (2020). Web Semantics : Science , Services and Agents on the World Wide Web Evaluating and comparing ontology alignment systems : An MCDM approach. *Web Semantics: Science, Services and Agents on the World Wide Web*, *64*, 100592. Retrieved from https://doi.org/10.1016/j.websem.2020.100592 doi: 10.1016/j.websem.2020.100592

Ontology Alignment Evaluation Initiative. (2020). *Results of evaluation for the conference track within oaei 2020.* Retrieved from http://oaei.ontologymatching.org/2020/results/conference/

Ontology Alignment Evaluation Initiative. (2021). *Ontology alignment evaluation initiative.* Retrieved from http://oaei.ontologymatching.org/

Pennington, J., Socher, R., & Manning, C. D. (2014). GloVe: Global vectors for word representation. *EMNLP 2014 - 2014 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference*(January 2014), 1532–1543. doi: 10.3115/v1/d14-1162

Rahm, E., & Bernstein, P. A. (2001). A Survey of Approaches to Automatic Schema Matching . A survey of approaches to automatic schema matching. (December). doi: 10.1007/s007780100057

Reimers, N., & Gurevych, I. (2020). Sentence-BERT: Sentence embeddings using siamese BERT-networks. *EMNLP-IJCNLP 2019 - 2019 Conference on Empirical Methods in Natural Language Processing and 9th International Joint Conference on Natural Language Processing, Proceedings of the Conference*, 3982–3992. doi: 10.18653/v1/d19-1410