Space mission design ontology: extraction of domain-specific entities and concepts similarity analysis

Audrey Berquand *, Yashar Moshfeghi[†], Annalisa Riccardi [‡]
University of Strathclyde, Glasgow, United Kingdom

Expert Systems, computer programs able to capture human expertise and mimic experts' reasoning, can support the design of future space missions by assimilating and facilitating access to accumulated knowledge. To organise these data, the virtual assistant needs to understand the concepts characterising space systems engineering. In other words, it needs an ontology of space systems. Unfortunately, there is currently no official European space systems ontology. Developing an ontology is a lengthy and tedious process, involving several human domain experts, and therefore prone to human error and subjectivity. Could the foundations of an ontology be instead semi-automatically extracted from unstructured data related to space systems engineering? This paper presents an implementation of the first layers of the Ontology Learning Layer Cake, an approach to semi-automatically generate an ontology. Candidate entities and synonyms are extracted from three corpora: a set of 56 feasibility reports provided by the European Space Agency, 40 books on space mission design publicly available and a collection of 273 Wikipedia pages. Lexica of relevant space systems entities are semi-automatically generated based on three different methods: a frequency analysis, a term frequency-inverse document frequency analysis, and a Weirdness Index filtering. The frequency-based lexicon of the combined corpora is then fed to a word embedding method, word2vec, to learn the context of each entity. With a cosine similarity analysis, concepts with similar contexts are matched.

I. Nomenclature

AI	=	Artificial Intelligence
BNC	=	British National Corpus
$CBOW$	=	Continuous Bag-of-Words
CDF	=	Concurrent Design Facility
CE	=	Concurrent Engineering
DEA	=	Design Engineering Assistant
$ECSS$	=	European Coordination for Space Standardization
ES	=	Expert System
ESA	=	European Space Agency
F	=	Frequency
IR	=	Information Retrieval
NLP	=	Natural Language Processing
NLU	=	Natural Language Understanding
OL	=	Ontology Learning
PCA	=	Principal Component Analysis
PoS	=	Part of Speech
SG	=	Skip-Gram
$TF - IDF$	=	Term Frequency - Inverse Document Frequency
WI	=	Weirdness Index

*PhD student, Mechanical and Aerospace Engineering Department, audrey.berquand@strath.ac.uk, AIAA Student Member

[†] Lecturer, Computer And Information Sciences Department, yashar.moshfeghi@strath.ac.uk

[‡] Lecturer, Mechanical and Aerospace Engineering Department, annalisa.riccardi@strath.ac.uk

II. Introduction

Accumulated unstructured and semi-structured data on space mission design contain a wealth of information for experts involved in the early stages of space mission design. Knowledge reuse is essential to kick-start a study and correctly estimate the initial values that will allow the design to converge. However, it can quickly become highly time-consuming to search through the non-neglectable amount of data accumulated over the past decades. As found in [1], experts involved in feasibility studies can spend up to 50% of their work time searching for information. Information Retrieval (IR) methods combined with the recent advances of Natural Language Processing (NLP) and Understanding (NLU), should allow experts to access information more quickly, more efficiently and overcome the current 'IR bottleneck'.

ESs are computer programs able to capture human expertise and mimic experts' reasoning. An ES's basic architecture relies on a knowledge base, an inference engine, and a User interface. The Design Engineering Assistant (DEA) is an Expert System (ES), designed to support the engineers in the conceptual design phase of new missions. The DEA aims to considerably reduce the time spent to access information on past missions. The high-level architecture of the DEA, requirements and the result of a set of interviews involving ESA experts are presented in previous publications [1], [2]. To our best knowledge, the Daphne virtual assistant, in development at Texas A&M University and presented in [3], is the most similar concept to the DEA. However, Daphne focuses on Earth Observation missions, and builds its knowledge on a manually defined ontology and structured database.

To organize information within the knowledge base, the DEA needs to understand what are the concepts characterizing a space mission, as well as, the relationships interconnecting these concepts. To automatize and accelerate the process, the DEA has to identify these concepts by itself within a large collection of documents related to space mission design. In other words, the DEA is semi-automatically generating the foundations of a space mission design ontology. The generation of an ontology, its maintenance and enrichment are lengthy processes which require the involvement of several domain experts. There is currently no single European Ontology for Space projects although discussions were kick-started by ESA in June 2019 [4]. First introduced by [5], Ontology Learning (OL) is a recent field of research which encompasses the set of methods and techniques to build an ontology in a semi-automatic fashion. Several OL approaches rely on the OL Layer Cake involving theoretical consecutive steps to achieve OL [6], [7]. A diagram of the OL Layer Cake is shown in Fig.1. This paper focuses on the implementation of the first two layers, 'Terms' and 'Synonyms'.

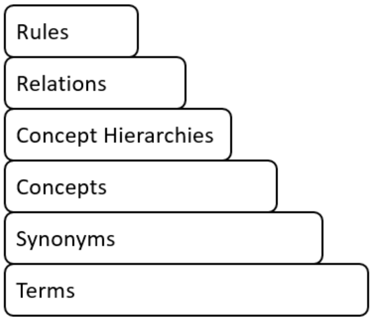

Fig. 1 Ontology Learning Layer Cake based on [7]

The study focuses on three corpora: feasibility reports provided by the ESA Concurrent Design Facility (CDF), books related to space mission design and a collection of Wikipedia pages on spacecraft design, all presented in section III. The methodologies used to automatically identify and evaluate candidate entities as well as extract synonym concepts are presented in section IV. Finally, the results obtained are discussed in section V.

III. Corpora

The methods presented in this study were applied to three corpora: feasibility reports provided by the ESA CDF, open-source books related to space mission design and a collection of Wikipedia pages on spacecraft design, used in a previous study [8]. Each corpus document, usually in .pdf or .doc format, is first parsed locally with the Apache Tika library [9], and saved as json files.

A. Wiki Corpus

The Wikipedia corpus is based on the corpus developed for a Latent Dirichlet Allocation Topic Modeling study presented in [8]. The Wikipedia page on Spacecraft Design was used as a starting point to find additional space mission design related content, using the hyperlinks interconnecting the web pages and the Python Selenium library. From the thousand web pages found with the automatic scrapping, 273 were manually selected to form the 'Wiki' corpus.

B. Feasibility Reports Corpus

The second corpus is a collection of proprietary feasibility studies reports provided by the ESA CDF team. This collection is composed of 56 reports spanning from 2000 to 2018, and includes a wide range of missions, from Earth Observation to Lunar missions. The original reports are not public and were made available for this study via a partnership. The CDF facility is based at the technical center of ESA, ESTEC, in The Netherlands. Methods of Concurrent Engineering (CE) are applied to space missions feasibility studies and pre-phase A. CE involves the simultaneous participation of all main disciplines related to the design of a project, working in parallel in the same facility. CE has proven to be a key asset in improving the quality of studies' outputs [10].

C. Books Corpus

The third corpus is formed by 40 books related to space mission design, and which content are available publicly. The selected books represent several fields and sub-fields of space mission design as summarised in Fig.2.

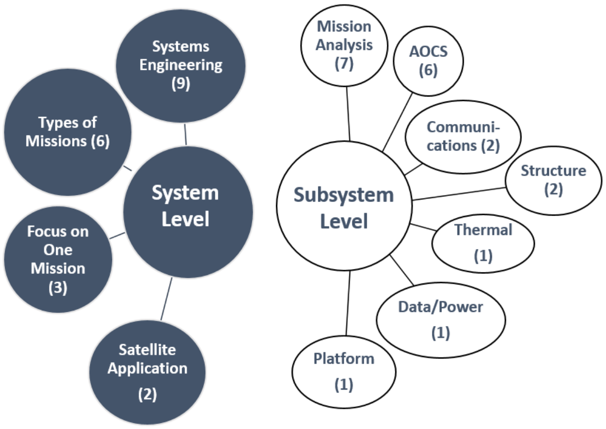

Fig. 2 Main topics of books selected for the Book corpus

IV. Methodology

A. Natural Language Processing Pipeline

Natural Language Processing (NLP) is a branch of Artificial Intelligence (AI) which enables computers to understand, interpret, process and manipulate human (natural) language. Natural language is by nature complex, ambiguous and highly context-dependent. A robust NLP pipeline is essential to preprocess the outputs of the raw text extraction before attempting to identify entities. Each corpus is preprocessed separately with the same pipeline. The NLP pipeline is based on the Natural Language Tool Kit (NLTK) Python library.

First, each corpus document is tokenized. Numerical tokens, non-English characters, and urls are removed. Acronyms are expanded based on the European Coordination for Space Standardization (ECSS) list of abbreviated terms [11]. Since the current NLP Pipeline lacks disambiguation, acronyms with several possible expansions are not expanded. Acronyms corresponding to named entities such as 'ESA', 'CERN' or 'NASA' are not expanded. The tokens are normalised before scanning for multi-words. The tokens corresponding to multi-words contained in the ECSS glossary of terms [12] are replaced within the corpora as one token containing the multi-word. The ECSS multi-words list is completed with a list of extra multi-words validated by domain experts. Lemmatization is applied to the tokens to prevent grammatical redundancy. Remaining basic English stop words, as well as punctuation, are removed. A term frequency-inverse document frequency (tf-idf) analysis of each corpus is run to identify the tokens with the lowest score. The 15% of tokens with the lowest tf-idf are removed as tokens with low tf-idf have low informativeness value. Therefore a tailored stop word list is generated for this study. Table 1 provides further information on the corpora statistics following preprocessing. Table 2 provides an insight into the most common multi-words and acronyms found in each corpus.

Table 1 Corpora Statistics

Corpus	Wiki	Feasibility Reports	Books	Combined Corpus
Number of documents	273	56	40	369
Total Number of tokens	629,603	1,463,775	2,951,800	5,045,178
Corpus Size	9.6 MB	542 MB	986 MB	1.5 GB
Average tokens number per document	2,306	26,139	73,795	N/A
Corpus Dictionary size	32,320	32,825	85,573	115,111
Corpus Dictionary size including only nouns	15,198	14,292	36,578	49,899

Table 2 Most common acronyms and multi-words found in each corpus

Corpus	Wiki Corpus	Feasibility Reports	Books
Top 5 multi-words	magnetic field deep space solar power low earth orbit solar cell	solar array propulsion system dry mass ground station data rate	remote sensing ground station magnetic field launch vehicle thermal control
Top 5 acronyms	GPS (Global Positioning System) UTC (Universal Time Coordinated) DRAM (Dynamic Random Access Memory) PV (Pressurized Pressure Vessel) SI (international system of units)	S/C (spacecraft) TRL (technology readiness level) AOCS (attitude and orbit control system) RF (radio frequency) GNC (guidance navigation and control)	GPS LEO (Low Earth Orbit) GEO (Geostationary Orbit) OBC (On-board Computer) dB (decibel)

B. Identification of candidate entities

Candidate entities are limited to 'nouns', excluding all other part-of-speech (PoS) types such as 'verbs', 'adjectives', or 'adverbs'. Using the NLTK POS tagger, only the tokens associated with a 'NN' (noun singular) or 'NNS' (noun plural) tags are kept. 'NNP' (proper noun singular) and 'NNPS' (proper noun plural) are filtered. The tokens other than nouns are only filtered for the 'Terms' layer step, all tokens will be used for the word2vec modeling. Figure 3 displays the PoS tags distribution for each corpus. The trend of tags is interestingly similar for the Wiki, Reports and Books corpora. To compare to a large general British corpus, the PoS tags for the British National Corpus (BNC)[13] is added to the figure. The representation of adjectives appears to be doubled in the study corpora compared to a classic English corpus.

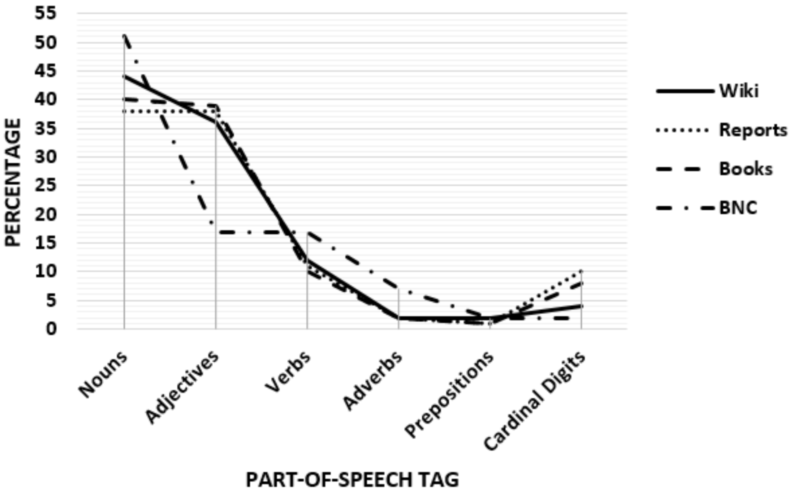

Fig. 3 Distribution of the main types of Part-of-Speech tags for study corpora and BNC

A common first step to automatically identify a corpus lexicon is to extract its most frequent words. The assumption is that a document related to a specific topic will more frequently use domain-specific words. If frequent words are more likely to represent the concepts of a corpus, it is however not enough to ensure that the candidate entities are domain-specific and not just common words frequently used in English. To distinguish the domain-specific terms, two methods based on Term Frequency - Inverse Document Frequency (tf-idf) and the Weirdness Index (WI) [14] are implemented . This methodology yields three lexica of candidate entities per corpus: a frequency-based lexicon, a frequency-based with tf-idf filtering lexicon, and a frequency-based with Weirdness Index filtering lexicon. A final set of lexica is also generated combining all three corpora.

1) **Lexicon 1: Term Frequency**
   This lexicon includes all nouns with a frequency above the tokens' frequency average.

2) **Lexicon 2: Term Frequency - Inverse Document Frequency (tf-idf) Filtering**
   This lexicon is built upon the first lexicon, filtering tokens with tf-idf scores below average. Tf-idf is a well-known information retrieval and text mining method [15]. The tf-idf weight is the multiplication of two terms, the term frequency, tf, and the inverse document frequency, idf. It is used to distinguish the terms with high frequencies in a specific corpus but with low frequencies over collections of documents. The idf weight of a term j is computed as follow:

$$idf_j = \log \frac{D}{d_j} \tag{1}$$

   where D is the number of documents in the corpus, and d_j is number of documents containing the term j.
   In the NLP pipeline, the tf-idf scores are used to filter the terms with low tf-idf scores and therefore low informativeness value, while in the entity identification steps, the terms with a tf-idf above average are kept.

3) **Lexicon 3: Weirdness Index (WI) Filtering**
   This second lexicon is built upon the first lexicon and by filtering tokens with a Weirdness Index (WI) below average. This method is presented in [14] applied to a nuclear physics corpus. This index allows to compare the use of a word, based on its frequency, between a domain-specific corpus and a large corpus representing the general language. In this case, the latter corpus used is the BNC. The index, W, is calculated as follow:

$$W = \frac{N_G f_S}{(1 + f_G) N_S} \tag{2}$$

   where f_S is the frequency of the word in the specialized corpus, f_G its frequency in the general corpus, the BNC, and N_S, N_G are respectively the number of tokens in the specialized and general corpus.

C. Validation of candidate entities

As mentioned previously, there is currently no official European Space Systems ontology. To evaluate the validity of the candidate entities automatically extracted, a comparison is run against a general English lexicon and a domain-specific dictionary.

1) **Comparison to a general English lexicon: WordNet.** WordNet is an open-source large lexical database of English, developed by the Princeton University [16]. Calling Wordnet a 'lexicon', a list of words, is a misuse of language as the authors developed a tool that resembles a thesaurus, gathering similar concepts into synsets. WordNet is however used as a mere lexicon for this validation step. The assumption is that classic English thesauri such as WordNet are usually not adapted to domain-specific lexicon. Therefore, to represent a domain-specific lexicon, the final lexicon of candidate entities should minimise the number of entities also found in WordNet.

2) **Comparison to a domain-specific dictionary: ECSS Terms and Accronyms.** The ECSS glossary of terms and definitions is a human validated dictionary of terms related to space systems [12]. This glossary is already partly used in the NLP pipeline to identify multi-words. Adding the 1-grams to the multi-words list, it forms a human validated space systems lexicon. The ECSS list of Abbreviated terms [11] complements the multi-words lexicon for this validation steps. The final lexicon of candidate entities automatically extracted should maximise the number of entities also found in the ECSS documents.

D. Merging of similar candidate entities

The methodology of the Term Layer presented above yields a list of domain-specific terms, candidate entities for a space systems ontology. Some candidate entities can be associated with similar concepts, for instance, "satellite" and "spacecraft". Authors may use one concept or another depending on their background or personal preferences. Being able to merge similar entities allows to identify new connections between concepts that may have been overlooked otherwise. Available British thesauri such as WordNet can provide a basis for similar concept identification. However, these thesauri are usually not adapted to domain-specific lexicon. Therefore in the case of specific topics, new synsets must be developed. A main assumption for the identification of similar concepts supported by [17] and [18] is that similar terms are found in a similar context. Word embedding methods, such as word2vec, allow to map the context of a term into a vector. The embedding captures semantic correlations which then allows to identify synonym concepts via similarity metrics, such as the cosine similarity.

1) **Word2vec:**
Word2vec is a word embedding method proposed by Mikolov, et al, in 2013 [19]. Since then, word2vec has been widely applied in the literature [20], [21]. Word2vec is a two-layer shallow neural network used to learn the embedding of words by exploiting word co-occurrence in a contextual window. Words sharing common contexts should be close in the semantic space. The two main modeling methods of word2vec are the skip-gram (SG) and continuous bag-of-words (CBOW) methods. With the SG architecture, the target word predicts its surrounding words, or context. In a CBOW architecture, the surrounding words are used to predict the target word. Negative sampling or hierarchical softmax are used to train the word2vec model. According to [20], negative sampling yields better results for frequent words than hierarchical softmax. In this paper, both CBOW and SG architectures will be tested, with a negative sampling training model. The models are generated with the Python Gensim library [22].

2) **Cosine Similarity:**
To find terms with close contexts, the cosine similarity is computed in-between the context vectors provided by the word2vec models. The cosine similarity $cos(\theta)$, between two vectors A and B, of dimension n, is computed as follow [21]:

$$cos(\theta) = \frac{A.B}{\mid A \mid\mid B \mid} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}} \tag{3}$$

Table 3 10 top and bottom tokens of the three corpora's lexica

Corpus	Wiki			Feasibility Reports			Books		
Method	F	TF-IDF	WI	F	TF-IDF	WI	F	TF-IDF	WI
Top 10 entities	space	voyager	magnetic field	mission	differential scanning calorimetry	technology readiness level	system	themis	global positioning system
	system	battery	global positioning system	mass	lander	attitude and orbit control system	orbit	on board computer	low earth orbit
	orbit	kosmos	flyby	design	water for injection	solar array	space	power control and distribution unit	geostationary orbit
	spacecraft	orbit	dynamic random access memory	system	tug	radio frequency	spacecraft	recurrence	on board computer
	energy	cell	low earth orbit	spacecraft	mechanism analytical verification	guidance navigation and control	time	sar	power control and distribution unit
	earth	transistor	deep space	requirement	rover	perigee	data	substorm	ground station
	time	radiation	pressurized pressure vessel	power	mirror	data handling system	mission	cruising	downlink
	mission	memory	international system of units	table	telescope	geostationary orbit	design	load	radio frequency
	power	capacitor	ranging tone frequency	esa	cubesats	multi layer insulation	power	herschel	global navigation satellite system
	link	mission	complementary metal oxide semiconductor	risk	impactor	propulsion system	phase	mothercraft	electric propulsion
Bottom 10 entities	irradiance	sense	kari	ethylene oxide	generator	prisma	activation energy	sound	myr
	multi layer insulation	blue	radioisotope	pdd	intermediate frequency	huygens	elasticity	valve	proba
	intra vehicular activity	transport	photovoltaics	expertise	specie	dlr	emittance	cabin	vsat
	leo	orbital debris	thruster	vacuum ultraviolet	burn	thermistor	configuration management	communications link transmission unit	multiplexer
	multi-layer	introduction	inverter	depth of discharge	snecma	libration	fallback	hydrogen	quantization
	interconnection	glass	ganymede	modal	gaussian minimum shift keying	callisto	firmware	removal	tradeoff
	scenario	distribution	electrodynamics	success criteria	maximum power point tracker	kiruna	timeliness	trend	phobos
	trapping	transformer	gyroscope	sense	closure	actuation	ellipso	product assurance	microsatellite
	attenuation	place	dosimetry	dlr	processor	actuator	apex	diode	themis
	designer	jaxa	cosmic-ray	surveillance	rosetta	emissivity	fpgas	maturity	sso
Lexicon Size	1,839	563	196	1,388	395	236	2,789	919	423
Threshold	22	88.35	29	57	43.4	75	43	57.1	30

V. Results

The results are first generated for each corpus separately before combining them. For each corpus, three lexica containing candidate entities are generated, based on a frequency-only analysis, a frequency analysis followed by a tf-idf filtering, and a frequency analysis followed by a Weirdness Index filtering. Only the tokens tagged as nouns can be considered as candidate entities. Word2vec word embedding and cosine similarity are applied to the frequency lexicon of the combined corpora to identify similar concepts.

A. Generation of candidate entities: Term Layer

Table 3 displays, for each corpus, the lexica generated either from the frequency analysis (labelled ’F’), using a tf-idf filtering (labelled ’TF-IDF’), or using a WI filtering (labelled ’WI’). For each lexicon, only the top 10 and bottom 10 entities are displayed. The lexica’ sizes and thresholds are indicated in the last table line. The three ’WI’ lexica top entities are mostly complex words, 2-grams or more, such as ’technology readiness level’ or ’dynamic random access memory’. The ’F’ lexica essentially contain 1-gram words. Similar concepts seem to be highlighted in the top 10 of each ’F’ lexicon. Concepts such as ’spacecraft’ or ’mission’ appear in all three frequency-based lexica.The tf-idf lexica provides more diversity by underlining domain-specific terms less complex than the terms highlighted by the ’WI’ lexica but less common than the top entities of the ’F’ lexica. These observations lead to the conclusion that a frequency-based method can be used to identify fundamental and basic concepts, while the tf-idf and WI methods can respectively highlight corpus-specific and complex concepts. Finally, the quality of the 10 bottom entities illustrates that simple thresholds based on the frequency, or WI average, are not restrictive enough to filter non-relevant concepts such as ’trend’ or ’place’.

Table 4 displays the top and bottom candidate entities for a corpus combining all three corpora. The trends of the basic concepts being represented by the ’F’ lexicon, and complex concepts being highlighted by the ’TF-IDF’ and ’WI’ lexica appear to be also valid for the combined corpora. The number of candidate entities increases with the initial size of corpus dictionaries, therefore, using a more complete and diversified corpus should ensure that no concept is missed. The bottom entities results stress the need for human annotators to validate the semi-automatically extracted entities, notably to set the threshold to filter non-relevant entities, which might vary depending on the initial corpus. The ’F’ lexicon of the combined corpora will be used as the input to the Synonym layer as it is the largest lexicon.

Table 4 10 top and bottom tokens of the combined corpora’s lexica

Corpus	Combined Corpora						
method	F	TF-IDF	WI	method	F	TF-IDF	WI
Top 10 entities	satellite	requirement	low earth orbit	Bottom 10 entities	bipod	laboratory	endeavor
	system	mission	global positioning system		skynet	cebreros	tdma
	orbit	satellite	technology readiness level		high-throughput	balance	dayside
	spacecraft	orbit	geostationary orbit		crystalline	duty	thermistor
	mission	esa	attitude and orbit control system		viscosity	office	dlr
	space	design	radio frequency		legislation	timeline	linearization
	design	data	solar array		competitor	echo	cdma
	time	risk	downlink		spreadsheet	myriade	traceability
	data	spacecraft	ground station		aquarius	collection	imagers
	mass	phase	on board computer		vanguard	transducer	non-space
Lexicon size	3,307	904	591	-	-	-	-
Threshold	55	333.13	33	-	-	-	-

B. Entities Validation with pre-existing lexica

Figures 4 and 5 respectively display, for each lexicon type and each corpus, the percentage of tokens also found in WordNet or in the ECSS glossary of terms and acronyms. Once again, the lexicon based only on the frequency analysis is labelled ’F’, the lexicon using a tf-idf filtering is labelled ’TF-IDF’, and the lexicon using WI filtering is

labelled 'WI'. As described previously, the assumption is that a valid space system set of terms would minimise the number of common tokens with a general lexicon such as WordNet and maximize the number of common tokens with a domain-specific lexicon such as the ECSS glossary of terms and acronyms.
Based on Fig.4, the percentages of common tokens found both in the corpora and WordNet are rather high for all frequency and TF-IDF lexica. The lexicon based on a Weirdness Index filtering ('WI'), have noticeably less in common with the WordNet lexicon. The 'WI' lexicon of the Wikipedia corpus, which is based on Wikipedia pages addressed to a general public and less likely to contain technical terms scores the highest. The WI method appears to be the most efficient to identify domain-specific terms.

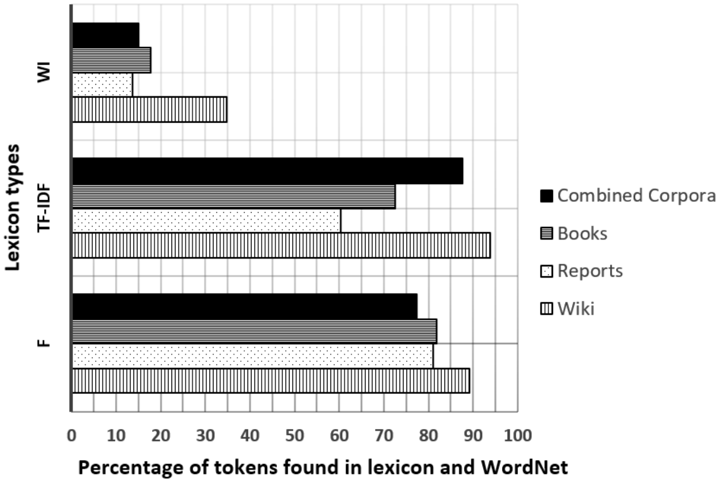

Fig. 4 Comparison of tokens found both in lexica and WordNet, for each corpus

Based on Fig.5, the 'WI' lexica have the highest matching percentages. The commonality peak, 47.5%, is reached for the reports lexicon based on ESA documents. These reports are the most likely to adhere to the European standards, and therefore to include ECSS terms and acronyms.

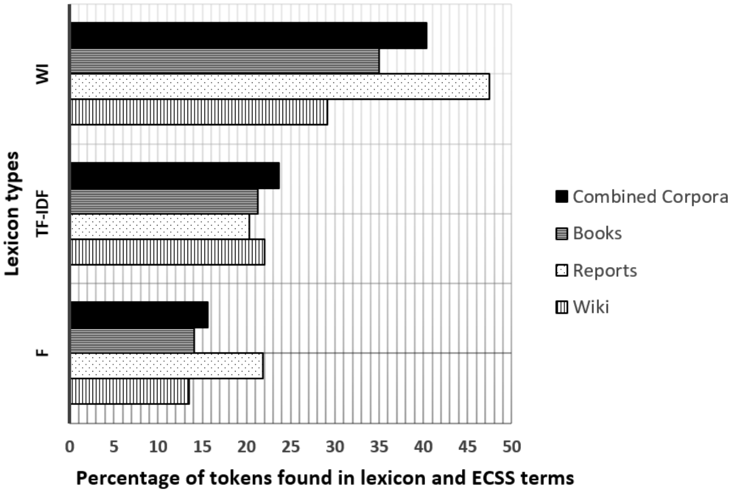

Fig. 5 Comparison of tokens found both in lexica and the ECSS terms and acronyms, for each corpus

C. Identification of Similar Concepts: Synonym Layer

1) **Word2vec Parameter settings:**
The dimension of the embedding is set to 100, and the number of negative sampling to 5. Window sizes from 2 to 18 were tested for all CBOW and SG models. Eventually, the window size was set to 6 for all models. Two models, a CBOW negative sampling model and a SG negative sampling model, were trained on the frequency lexicon of the combined corpora. The cosine similarity threshold was set to 0.9.

2) **Concepts Similarity Results:**
Out of the initial 3,307 entities composing the combined corpora frequency lexicon, the SG model yielded 472 entities with at least one similar concept, meaning, with a cosine similarity above the threshold. The CBOW embedding produced 630 entities with at least one similar concept. The models' results were combined and the pairs of similar concepts suggested by the word2vec and cosine similarity were manually evaluated.
The majority of these pairs appeared to belong to similar lexical fields rather than represent inter-exchangeable concepts, synonyms. Some concepts were even associated with their antonyms, to their acronym or acronym expansion, or their British or American equivalent spelling. A Principal Component Analysis (PCA) is used to project the word embedding results in two dimensions. Similar concepts should have similar PCAs and therefore appear in the same area of the graph.

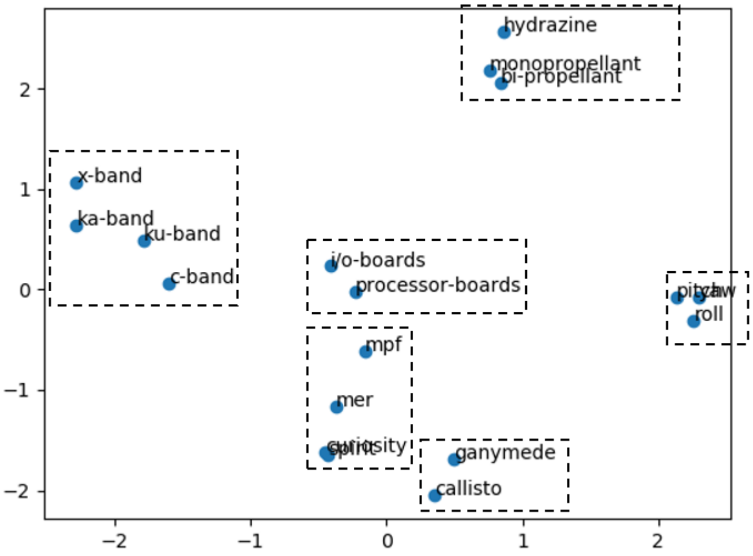

Fig. 6 PCA projection of similar concepts. (Bottom-center overlapping words are 'spirit', and 'curiosity'; Right-center overlapping words are 'pitch' and 'yaw')

Figure 6 displays a sample of similar concepts, forming 6 clusters. For clarity, dotted-line boxes are drawn around similar concepts clusters, highlighting, for instance, a 'propulsion' cluster including the concepts of 'hydrazine', 'monopropellant', and 'bi-propellant'. Other clusters could be manually labelled as 'bandwidth' (including 'x-band', 'ka-band', 'ku-band', and 'c-band'), 'mars rovers' (including 'mpf', 'mer', 'curiosity', and 'spirit'), 'axes' (including 'yaw', 'pitch', and 'roll'), 'processor' (including 'i/o-boards', and 'processor-boards') and 'Jupiter's Moon' (including 'ganymede', and 'callisto').

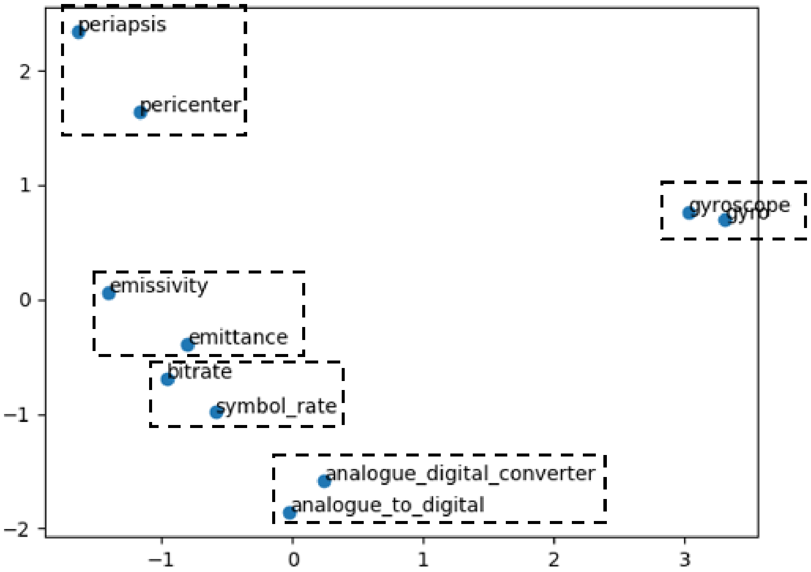

Fig. 7 PCA projection of entities embedded vectors representing synonyms concepts. (Middle right words are 'gyro' and 'gyroscope')

A fraction of concepts, around 2.5%, were paired with synonym concepts. A sample of these synonyms is displayed in Figure 7. Figure 8 displays samples of similar concepts corresponding to antonyms. The results presented in this section proved that word2vec and cosine similarity can successfully identify and associate similar concepts. However, domain-experts are needed to assess the type of context similarity which is being highlighted: similar lexical field, synonym, antonyms, acronyms, or spelling disparity. The samples presented in the Figures 6, 7, and 8 were all manually selected by a human annotator.

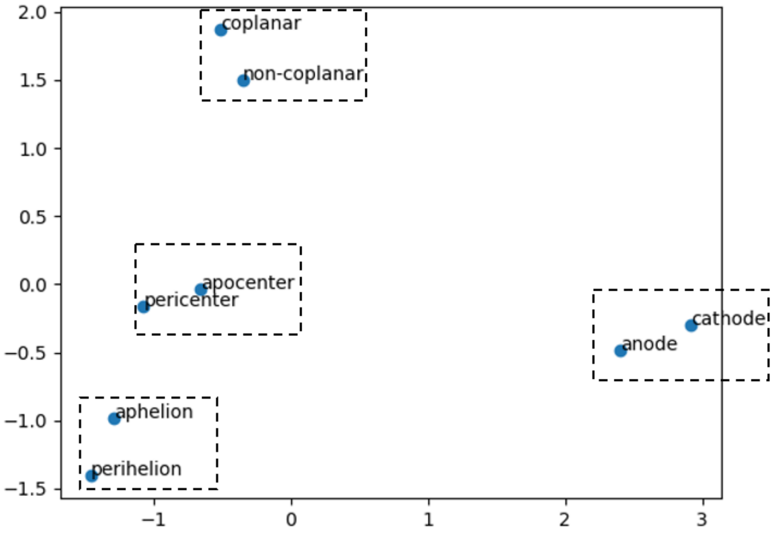

Fig. 8 PCA projection of entities embedded vectors representing antonyms concepts

VI. Conclusion and Next steps

This paper presented, to the best of our knowledge, the first application of the Ontology Learning Layer cake to a space mission design corpora. The study focused on the implementation of the two first layers of the OL Layer Cake, the Terms and Synonym layers. The corpus was formed of three corpora: a Wikipedia corpus, a set of ESA feasibility study reports, and books related to space mission design. The methodology for the Terms layer, relying on frequency analysis, complemented with a tf-Idf or a Weirdness Index filtering, allowed to semi-automatically identify key concepts in space systems engineering. The Weirdness Index filtering was found especially relevant for singling out complex concepts. The outputs of the Terms layer were compared to a general English lexicon, WordNet, and a domain-specific dictionary, the ECSS glossary of terms and acronyms. Word2vec models were trained on the frequency lexicon of the combined corpora, leading to the discovery of 630 entities sharing similar contexts with other entities. The concepts' similarities were manually verified by a human annotator. Most of the paired concepts belonged to similar lexical fields rather than represented interchangeable concepts. This paper has demonstrated the potential of the Ontology Learning Layer Cake for laying the basis of a space systems engineering ontology. Human domain-experts are however still needed to validate and select the entities and their synonyms. The NLP pipeline, the models and the wiki, and books corpora are available at `https://github.com/strath-ace/smart-nlp`.

The corpora' diversity and size were key to ensure the variety of entities identified. Due to the lack of available standard corpus collection in the space field, a first 'Space Mission Design' Test Collection is currently being assembled by the authors, allowing to discover more entities and concept similarities. As a future step, the authors are considering the implementation of a BERT (Bidirectional Encoder Representations from Transformers), a powerful language representation model, which could enhance the discovery of synonyms concepts. Finally, for future work, the next layers of the OL Layer Cake could be implemented.

Acknowledgments

The development of the Design Engineering Assistant is done in the frame of a Networking Partnership Initiative (NPI) involving the European Space Agency (ESA), and the industrial partners RHEA Systems, Airbus and satsearch. The authors warmly thank Tiago Soares (ESA), Hans-Peter de Koning (ESA), Sam Generé (RHEA), Norber Brauer (Airbus) and Kartik Kumar (satsearch) for their valuable guidance, support, and expertise.

References

[1] Berquand, A., Murdaca, F., Riccardi, A., Soares, T., Generé, S., Brauer, N., and Kartik, K., "Artificial Intelligence for the Early Design Phases of Space Missions," *IEEE Aerospace*, IEEE, Big Sky, Montana, US, 2019.

[2] Berquand, A., Murdaca, F., Dr. Riccardi, A., Soares, T., Generé, S., Brauer, N., and Kartik, K., "Towards an Artificial Intelligence based Design Engineering Assistant for the Early Design of Space Missions Audrey Berquand," *69th International Astronautical Congress (IAC)*, IAF, Bremen, Germany, 2018.

[3] Viros, A., and Selva, D., "Daphne: A Virtual Assistant for Designing Earth Observation Distributed Spacecraft Missions," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing (J-stars)*, 2019. doi:10.1109/JSTARS.2019.2948921.

[4] Terraillon, J.-L., de Koning, H.-P., and Valera, S., "Overall System Modelling for System Engineering (OSMoSE) - Space System Ontology 1st Brainstorming Workshop Report," ESA, ESTEC, 2019.

[5] Maedche, A., and Staab, S., "Ontology Learning for the Semantic Web," *IEEE Intelligent Systems*, Vol. 2, No. 16, 2001, pp. 72–79. doi:10.1109/5254.920602.

[6] Staab, S., and Studer, R., *Handbook on ontologies*, 2^{nd} ed., Springer, 2009.

[7] Petasis, G., Karkaletsis, V., Paliouras, G., Krithara, A., and Zavitsanos, E., "Ontology Population and Enrichment: State of the Art," *Knowledge-Driven Multimedia Information Extraction and Ontology Evolution - Bridging the Semantic Gap*, 2011.

[8] Berquand, A., McDonald, I., Riccardi, A., and Moshfeghi, Y., "The automatic categorisation of space mission requirements for the Design Engineering Assistant," *70th International Astronautical Congress (IAC)*, IAF, Washington, D.C., US, 2019.

[9] The Apache Software Foundation, "Apache Tika 1.20," `https://tika.apache.org/1.20/index.html`, -.

[10] Bandecchi, M., Melton, B., Gardini, B., and Ongaro, F., “The ESA/ESTEC Concurrent Design Facility,” *2nd Concurrent Engineering Conference (EuSEC)*, Munich, Germany, 2000.

[11] ECSS, “ECSS Abbreviated Terms,” `https://ecss.nl/home/ecss-glossary-abbreviations/`, 2017.

[12] ECSS, “ECSS Terms and Definition,” `https://ecss.nl/home/ecss-glossary-terms/`, 2007.

[13] BNC Consortium, “British National Corpus,” `http://www.natcorp.ox.ac.uk/`, -.

[14] Ahmad, K., and Gillam, L., “Automatic Ontology Extraction from Unstructured Texts,” *On the Move to Meaningful Internet Systems 2005: CoopIS, DOA, and ODBASE*, edited by R. Meersman and Z. Tari, Springer Berlin Heidelberg, Berlin, Heidelberg, 2005, pp. 1330–1346.

[15] Rezgui, Y., “Text-based domain ontology building using Tf-Idf and metric clusters techniques,” *The Knowledge Engineering Review*, Vol. 22, No. 4, 2007, pp. 379–403. doi:10.1017/S0269888907001130.

[16] Princeton University, “About Wordnet.” `https://wordnet.princeton.edu/`, 2010.

[17] Siniakov, P., “Recognition of synonyms by a lexical graph,” *Proceedings of the Workshop on Adaptive Text Extraction and Mining (ATEM 2006)*, 2006. URL `https://www.aclweb.org/anthology/W06-2205`.

[18] Lund, K., and Burgess, C., “Producing high-dimensional semantic spaces from lexical co-occurrence,” *Behavior Research Methods, Instruments, and Computers*, Vol. 28, No. 2, 1996, pp. 203–208. doi:10.3758/BF03204766.

[19] Mikolov, T., Sutskever, I., Chen, K., Corrado, G., and Dean, J., “Distributed Representations of Words and Phrases and their Compositionality,” *Advances in neural information processing systems*, 2013, pp. 3111–3119.

[20] Zhang, L., Li, J., and Wang, C., “Automatic synonym extraction using Word2Vec and spectral clustering,” *36th Chinese Control Conference, CCC*, 2017, pp. 5629–5632. doi:10.23919/ChiCC.2017.8028251.

[21] Smywiński-Pohl, A., Wróbel, K., Lasocki, K., and Strzała, M., “Automatic Construction of a Polish Legal Dictionary with Mappings to Extra-Legal Terms Established Via Word Embeddings,” *17th International Conference on Artificial Intelligence and Law*, Montreal, Canada, 2019.

[22] Řehůřek, R., and Sojka, P., “Software Framework for Topic Modelling with Large Corpora,” *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, ELRA, Valletta, Malta, 2010, pp. 45–50. `http://is.muni.cz/publication/884893/en`.