SpaceLDA: Topic Distributions Aggregation from a Heterogeneous Corpus for Space Systems

Audrey Berquand[a,*], Yashar Moshfeghi[b], Annalisa Riccardi[a]

[a] *Intelligent Computational Engineering Lab, Mechanical and Aerospace Department, University of Strathclyde, 75 Montrose St., G11XQ, Glasgow, United Kingdom.*
[b] *Computer And Information Sciences Department, University of Strathclyde, 26 Richmond St., G11XH, Glasgow, United Kingdom.*

Abstract

The design of highly complex systems such as spacecraft entails large amounts of documentation. Tracking relevant information, including hundreds of requirements, throughout several design stages is a challenge. In this study, we propose a novel strategy based on Topic Modeling to facilitate the management of spacecraft design requirements. We introduce spaceLDA, a novel domain-specific semi-supervised Latent Dirichlet Allocation (LDA) model enriched with lexical priors and an optimised Weighted Sum (WS). We collect and curate the first large collection of unstructured data related to space systems, combining several sources: Wikipedia pages, books, and feasibility reports provided by the European Space Agency. We train the spaceLDA model on three subsets of our heterogeneous training corpus. To combine the resulting per-document topic distributions, we enrich our model with an aggregation method based on an optimised WS. We evaluate our model through a case study, a categorisation of spacecraft design requirements. We finally compare our model's performance with an unsupervised LDA model and with a literature aggregation method. The results demonstrate that the spaceLDA model successfully identifies the topics of requirements and that our proposed approach surpasses the use of a classic LDA model and the state of the art aggregation method.

*Corresponding author
Email address: audrey.berquand@strath.ac.uk (Audrey Berquand)

Keywords: Topic Modeling, LDA, spacecraft design, requirements, aggregation

1. Introduction

Experts involved in the early stages of space mission design can spend from 25 to 50% of their work time searching for information (Berquand et al., 2019). Requirements Management, the process of documenting, analysing and tracking requirements, is an essential but time-consuming task. Although Machine Learning (ML) methods are today commonly used for downstream applications of space-based measurements, they are still largely underused at the earlier stages of the spacecraft life cycle. This study argues that Topic Modeling (TM), an ML method used to identify, learn, and extract latent topics from a corpus of documents, could enhance the management of requirements in the space field, notably supporting their categorisation. TM has been previously suggested as a preferred method for building text representations by (Al-Salemi et al., 2017) and (Yun and Geum, 2020). (Sriurai, 2011) also demonstrated that TM was a more efficient method for building feature representations of texts than the Bag of Words approach. TM differs from classic classification approaches such as Random Forest (Karasu and Altan, 2019), which is based on decision trees as it relies on probabilistic distributions. Following a bottom-up approach, TM enables the identification of terms representing, in practice, key spacecraft subsystems in a heterogeneous corpus related to space systems. In this context, this study introduces spaceLDA, a novel domain-specific semi-supervised TM model enriched with an optimised Weighted Sum (WS), fine-tuned for the extraction of topics related to space systems.

To train the spaceLDA model, a heterogeneous training corpus is collected. With the lack of benchmark data set for space systems, we gathered our own training corpora based on Wikipedia pages, feasibility reports provided by the European Space Agency (ESA), and books. The corpora were processed through a Natural Language Processing (NLP) pipeline that we tailored to space systems

by incorporating space terminology standards. As the training corpus is heterogeneous, the spaceLDA model is independently trained on the three corpus subsets to avoid under-representing smaller corpora. To merge the topic distributions obtained from each model, we propose a novel aggregation approach based on an optimised WS. The training of the spaceLDA model was in addition enriched by human-validated lexical priors that we defined, encouraging the discovery of topics related to key spacecraft subsystems.

The spaceLDA model is evaluated with a case study addressing the categorisation of spacecraft design requirements. Requirements extracted from public ESA reports are submitted as unseen data to the spaceLDA model. The resulting aggregated topic distribution enables the association of each requirement with a spacecraft subsystem. The spaceLDA performances are compared to an unsupervised LDA model trained on the same heterogeneous training corpus. The performance of the WS aggregation method is compared to a state of the art aggregation method based on the Jensen-Shannon (JS) divergence. To summarise, this research makes the following contributions:

1. We provide a first curated text collection related to space systems.
2. We train a domain-specific LDA model, named spaceLDA, enriched with lexical priors and an optimised WS.
3. We demonstrate that the spaceLDA approach outperforms the unsupervised LDA model and a literature method for aggregating per-topic word distributions.
4. We illustrate a practical application of TM for Requirements Management.

The rest of the paper is structured as follows. Section 2 introduces the background, Section 3 the spaceLDA approach and Section 4 the training and case study corpora. Section 5 details the methodology, and Section 6 the results furthermore discussed in Section 7. The code and curated text collection, excluding the ESA feasibility reports, are available at `https://github.com/strath-ace/smart-nlp`.

2. Background

Today ML methods are commonly used for downstream applications of space-based measurements. However, they are still largely underused at the earlier stages of a spacecraft life cycle. In this section, a few examples of TM applications in the space field are explored. Several ML methods that could enhance Requirements Management are discussed. Finally, the method to train a semi-supervised LDA model with lexical priors is summarised.

2.1. Application of Topic Modeling to space systems

While TM has been commonly used for Text Mining tasks, such as collaborative filtering (Moshfeghi et al., 2011) or trend forecasting (Shiryaev et al., 2017, Park et al., 2018), applications in the space field are scarce. Based on the literature, the majority of these applications focuses on the classification of space-based measurement products. (Liénou et al., 2010) applies LDA to automatically annotate large high-resolution satellite images based on visual words extracted from the images. (Văduva et al., 2018) implements an LDA model for temporal analysis to trace the evolution of features from Synthetic Aperture Radar imaging. Finally, (Bahmanyar et al., 2018) trains a joint multimodal LDA (mmLDA) for the classification of multisensor satellite imaging, a heterogeneous data set. (Layman et al., 2016) is the closest study to the work presented in this paper. (Layman et al., 2016) applied LDA to identify topics and trends in NASA problem reports. These reports included textual descriptions of anomalies detected during testing and operations, as well as details on the resulting corrective actions. TM successfully enabled the authors to extract trends of reported anomalies from thousands of documents. From these previous studies found in the literature, the use of TM at the early design stages of a space mission for requirements categorisation appears as a novel application.

2.2. Requirements Management and Machine Learning

Requirements Management is the process of documenting, analysing and tracking requirements. It is, therefore, an essential process for large-scale and

complex projects. (Iqbal et al., 2018) recently surveyed the ML methods applied to Requirement Engineering, noticing an increasing effort to merge both fields. The classification methods mentioned by the authors mostly include classic approaches such as Support Vector Machines, Conditional Random Field Network and Naïves Bayes. The authors only briefly mentioned Topic Modeling and the more recent method of word embedding, word2vec. The latter word embedding method proposed by (Mikolov et al., 2013) enables the mapping of the context of a term into a vector. Both TM and word embedding appear as novel methods for Requirements Management, let alone Requirements Management in the space field. However, without detailed preliminary knowledge of words most likely to describe each spacecraft subsystem, a word embedding approach could not be directly applied. The TM approach was therefore preferred as it enabled the definition of per-topic word distributions for each spacecraft subsystem. Based on the analysis of the current state of the art, the training of a domain-specific LDA tailored to space systems and its application to Requirements Management addresses a knowledge gap.

2.3. Semi-supervised Latent Dirichlet Allocation

LDA was first introduced in (Blei et al., 2003) as a generative probabilistic model for discrete data collections. Within an LDA model, each document is a probability distribution over topics, and each topic is a probability distribution over words. The probability distribution of topics T among a corpus of documents can be defined as in (Moshfeghi et al., 2011):

$$p(M|\alpha,\beta) = \int p(\theta|\alpha)\left(\prod_{i=1}^{N}\sum_{z=1}^{T} p(z|\theta)p(w_i|\beta_z)\right)d\theta \tag{1}$$

where M is a document composed of N words w_i, z is a topic from a set of latent topics T, $p(z|\theta)$ is a multinomial distribution given by θ and followed by topic z, $p(w_i|\beta_z)$ is the probability that word w_i belongs to topic z given by β_z. β and α are the Dirichlet distribution parameters, respectively for the per-topic word distribution and for the per-document topic distribution. θ follows the hyperparameter α.

The LDA model can be trained in a semi-supervised fashion to guide the extraction of latent topics. The initial probability distribution of a word to belong to a topic, $p(w_i|\beta_z)$, is randomly set at the start of the modelling process. For a semi-supervised LDA, the probability of certain words belonging to a topic can be increased at the start of the process to influence the composition of the per-topic word distribution. The concept of inputting lexical priors, or seed words, into a model is presented in (Jagarlamudi et al., 2012). With the Gensim Python library developed by (Řehůřek and Sojka, 2014) and used to train the spaceLDA model, η, a matrix representing for each topic, the probability of each word to belong to it, can be provided to the model to impose the asymmetric priors over the word distribution.

3. Approach

The core approach of the spaceLDA model training is based on two key components:

1. Lexical priors to steer the topics extraction.
2. An optimised WS to aggregate models trained on different corpus subsets.

Lexical priors reflecting key spacecraft subsystems can influence the extraction of topics relevant to space systems. The priors' probabilities are set to 0.95 while the probabilities of the remaining words are set to 0. This approach entails a semi-supervised training of the spaceLDA model. The lexical priors selection is furthermore detailed in 5.4.

The spaceLDA approach proposes a novel method to aggregate topic distributions. The first method to aggregate results of models trained on heterogeneous data is usually to alter the basic architecture of the LDA model and combine the data sets during the model training. In (Chen et al., 2018), the authors combined the Author Topic Model, developed by (Rosen-Zvi et al., 2004), with LDA to form a Heterogeneous Topic Model. However, a simpler and preferred approach would not alter the classic LDA architecture. Several authors have investigated post-training aggregations, meaning a merging of the

per-document topic distributions. To aggregate models, a common first step is to identify similar topic distributions. (Blei and Lafferty, 2007) relies on a graph-based method. (Blair et al., 2020) compared the cosine similarity and the JS divergence to identify similar topics, proving the higher performance of the latter in creating more coherent topics. (Blair et al., 2020) furthermore implemented the aggregation of models trained on an unchanging corpus but with varying Dirichlet priors and topic numbers.

The scope of this study differs from previous work as we intend to combine models trained on a heterogeneous corpus to capitalise on the diversity of the space mission design data we have collected. Combining independent models has the main advantage that it requires no modification of the underlying architecture of the basic LDA models. Thus allowing the use of standard libraries such as the Gensim Python library. In (Schnober and Gurevych, 2015), the authors combine models pre-trained with various LDA parameters and subsets of the same corpus preprocessed with disparate methods. However, the authors do not aggregate similar topics together but rather consider that they form a large list of distributions. In this paper, we introduce the concept of weighting and optimising the topic distributions obtained from different models over the same unseen data with an optimised WS. Only in (Alsumait et al., 2008), the concept of weight matrix is mentioned but is adopted to regulate the influence of more recent inputs to update an online LDA model. We will compare our aggregation of per-document topic distributions based on an optimised WS with a state of the art method to aggregate per-topic word distributions based on the JS divergence.

Fig. 1 summarises the approach presented in this study to train the spaceLDA model. From our curated text collection, a spaceLDA model is trained in a semi-supervised fashion with lexical priors. To avoid eclipsing smaller data sets, the model is trained separately on each training set based either on Wikipedia pages, feasibility reports or books. A requirement is extracted from the case study corpus and submitted to the models. Each model yields a per-requirement topic distribution, highlighting its most salient topics. To converge towards a single

topic distribution, the distributions are aggregated based on an optimised WS.

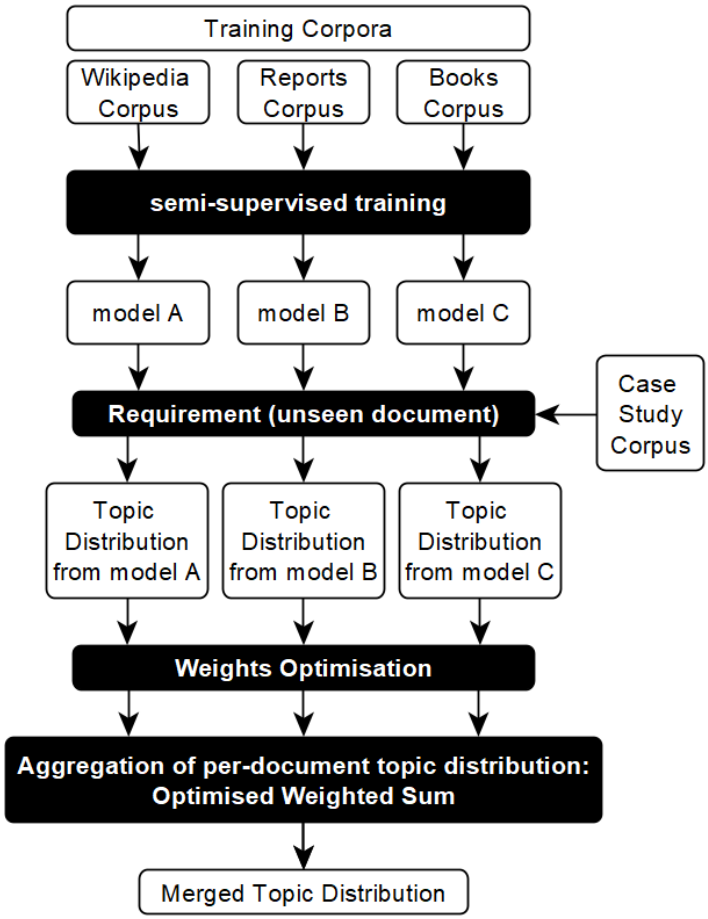

Fig. 1. Graphic representation of the spaceLDA approach

4. Corpora

4.1. Corpora introduction

The study relies on two types of corpora:

1. **The training corpus:** a collection of documents related to space systems, acquired from heterogeneous sources.

2. **The case study corpus:** a set of requirements extracted from public ESA documents.

The training corpus includes 273 Wikipedia web pages, 52 proprietary feasibility studies reports kindly provided by ESA, and 26 publicly available books. The size of the training data is 1.5 GB of raw textual data. However, since the training subsets have different sizes, they are used separately to train the spaceLDA model and prevent biases. The training corpora was carefully selected and filtered to target spacecraft subsystems and avoid introducing noisy topics. The content of each data set is furthermore detailed in the following paragraphs. Each corpus document was parsed with the Tika library developed by (The Apache Software Foundation, 2018).

4.2. Training corpora

To develop a domain-specific model, a domain-specific training set is an essential foundation. In this study, we use a novel text collection of unstructured data related to space mission design that we collected and curated.

4.2.1. Wikipedia corpus

The first corpus used to train the models is based on Wikipedia's freely available data. The Wikipedia page on spacecraft design (`https://en.wikipedia.org/wiki/Spacecraft_design`) was set as a starting point to find additional 'mission design' content, exploring the hyperlinks interconnecting the web pages. From the initial web page, six hyperlinks, judged as most relevant to a space mission corpus, were manually selected. These web pages were then automatically scrapped using the Python Selenium library, leading to the discovery of 1,023 additional non-redundant hyperlinks. The distribution of hyperlinks per web pages, including the main page on Spacecraft design, is shown in Fig. 2. The list of pages to be included in the corpus was manually filtered for relevance to the project scope and eventually yielded a corpus of 273 pages.

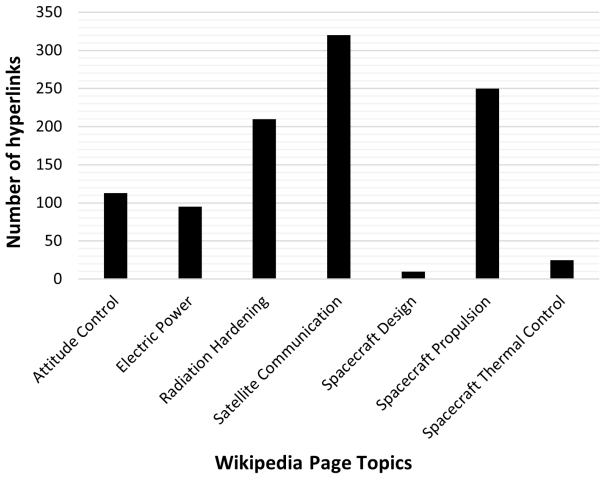

Fig. 2. Distribution of Wikipedia pages per spacecraft subsystems

4.2.2. Feasibility reports corpus

The second corpus is a collection of proprietary feasibility studies reports provided by the ESA Concurrent Design Facility (CDF) team. This collection is composed of 52 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. Only the chapters concerning the design of the spacecraft subsystems were used.

4.2.3. Books corpus

The third corpus contains 26 books related to space mission design, manually selected, and available publicly. The selected books represent several fields and sub-fields of space mission design, including classic textbooks such as (Larson and Wertz, 2005) and (Kapurch, 2007), as well as more subsystems-specific documents such as (Birur et al., 2003) and (Liu et al., 2019). The complete list can be found at `https://github.com/strath-ace/smart-nlp`.

4.3. Case study corpus: design requirements

Design requirements are usually associated with one spacecraft subsystem. The case study corpus includes 100 requirements extracted from two ESA documents, publicly available, the SMOS mission System Requirement Document (ESA, 2005) and MarcoPolo-R's Mission Requirement Document (ESA, 2012). The requirements within these documents are organised per subsystems. For instance, all power-related requirements are found under the chapter 'Power requirements'. Therefore, for each requirement, a subsystem to which the requirement belongs to can be extracted and used as ground truth in the case study. The distribution of requirements per topic is displayed in Fig. 3. From this corpus, 68 requirements related to the 7 topics of *Attitude and Orbit Control Subsystem* (*AOCS*), *Communication*, *Environment*, *On − Board Data Handling* (*OBDH*), *Power*, *Propulsion*, and *Thermal* are used to evaluate the models. Table 1 provides samples of requirements extracted from (ESA, 2005).

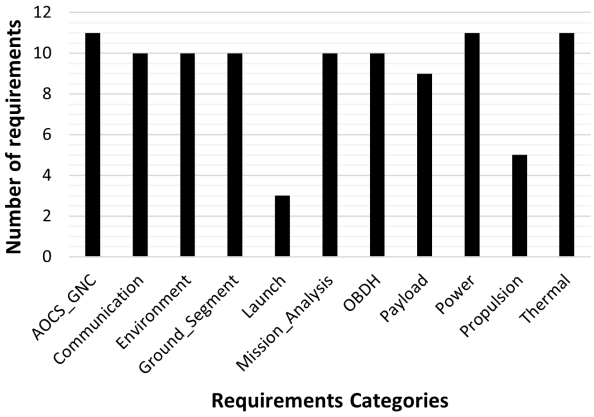

Fig. 3. Case study Corpus Distribution

Table 1. Sample of Design Requirements extracted from (ESA, 2005)

Subsystem	Requirement
Thermal	*'The thermal control shall be achieved by passive means and by heaters. The use of heat pipes shall be avoided.'*
Attitude and Orbit Control Subsystem	*'In Yaw Steering Mode, the attitude control laws shall be pre-defined law only dependant on one variable: True Latitude'*
Communication	*'The platform communications system shall provide the capabilities to transmit the data stream in S-band.'*

4.4. Corpora preprocessing

An NLP language pipeline based on the Python NLTK (Natural Language Toolkit) library by (Bird et al., 2009) is used to process both the training and case study corpora. Acronyms found in the corpora are expanded based on the European Coordination for Space Standardization (ECSS) list of abbreviated terms (ECSS, 2017). Tokens corresponding to phrases or multi-words found in the ECSS glossary of terms (ECSS, 2007) are replaced within the corpora as single tokens. The ECSS glossary of terms and definitions is a human-validated dictionary of terms related to space systems containing 2,106 unique words. The integration of the ECSS glossary of terms and acronyms into the NLP pipeline tailors it to space systems. A term frequency-inverse document frequency (tf-idf) analysis of each corpus is run to identify tokens with the lowest score. The 15% of tokens with the lowest tf-idf is added to the stop words list as tokens with low tf-idf have low informativeness value. Since the LDA modelling is influenced by the terms' frequency, the top 50 most frequent words of each corpus are manually verified. Table 2 provides an overview of the top 20 most frequent words for each training corpus. Despite the heterogeneity of their sources, the most frequent words are similar across the different corpora. Table 3 summarises the corpora' statistics post-processing.

Table 2. Top 20 most frequent words of each training corpus (organised by alphabetical order to underline similarities, words in bold are found in two lists, words in bold and underlined are found in all lists.)

Corpus	**Corpus I Wiki**	**Corpus II Reports**	**Corpus III Books**
Top 20 most frequent words	**antenna**, battery, cell, communication, **control**, **data**, electric, force, **launch**, magnetic, material, momentum, navigation, **orbit**, orbital, **power**, **propulsion**, radiation, radio, rocket	**antenna**, configuration, **control**, **data**, earth, instrument, **launch**, manoeuvre, mechanism, operation, **orbit**, panel, **payload**, **power**, propellant, **propulsion**, structure, tank, **thermal**, thruster	angle, **antenna**, attitude, **control**, **data**, function, ground, **launch**, motion, **orbit**, **payload**, performance, phase, **power**, sensor, temperature, test, **thermal**, vector, velocity

Table 3. Corpora' statistics post-processing

Corpus	**Corpus I Wiki**	**Corpus II Reports**	**Corpus III Books**	**Case study Corpus**
Number of documents	273	52	26	100
Number of tokens	507,222	598,407	1,378,037	1,457
Corpus Size	9.6 MB	542 MB	986 MB	17 KB
Average number of tokens per document	2,300	11,507	57,418	15
Dictionary Size	30,556	15,935	56,589	577

5. Methodology

5.1. Hyperparameters study

Three main inputs are required to train a model with the Python Gensim Library of (Řehůřek and Sojka, 2014):

- the *dictionary*, which maps words, or tokens, to identification numbers
- the *corpus*, or *document-term matrix*, which provides per document, the words identification numbers and their frequency within the document
- the *number of latent topics* to be defined after optimisation

The Dirichlet prior alpha is set to $1/n$ where n is the number of topics. The number of passes is set to 500. The first two inputs are derived from the corpus. To determine the number of latent topics, several spaceLDA models with different numbers of topics are trained with the Gensim Python library and compared. The evaluation metric of perplexity, presented in the next paragraph, determines which model is best fitted to represent the corpus' topic distribution. The training corpus is split between a training and a testing set, following the classic 80%/20% partition. 5-fold cross-validation is applied to find the number of optimal topics and retain the final model. The testing set is used for the final evaluation of the retained model post-optimisation.

5.2. SpaceLDA model evaluation

Perplexity is an intrinsic evaluation metric used to evaluate LDA topics (Blei et al., 2003, Shiryaev et al., 2017). Perplexity evaluates how well the probability distribution generated represents the corpus and measures the likelihood that the model will perform well with unseen, new, data. The value of perplexity must be minimised. Based on (Blei et al., 2003), perplexity over a test corpus, $per(D_{test})$ is expressed as:

$$per(D_{test}) = exp\left(-\frac{\sum_{d=1}^{M} \log p(w_d)}{\sum_{d=1}^{M} N_d}\right) \tag{2}$$

with M, the number of documents in the test sample, w_d the words in document d, $\log p(w_d)$ the log likelihood of document d, and N_d, the number of tokens in document d.

5.3. Topics labelling

The latent topics' word distributions produced by the model are not labelled. Therefore, we provide human-validated labels to ensure the reproducibility of the results. Three human annotators were involved in assigning topic labels to the word distributions, working independently and manually. A final label was elected when at least two of the three annotators agreed. Without a clear majority, the human annotators shortly debated to converge towards a single label. The annotators were given the following labels to choose from: *AOCS*, *Communication*, *Environment*, *Ground Segment*, *Launch*, *Mission Analysis*, *OBDH*, *Payload*, *Power*, *Propulsion*, and *Thermal*. The annotators also had the option to propose a topic label outside of this selection. It was made clear to the annotators that they could associate more than one label to each word distribution and that one label could be associated with several distributions.

5.4. Lexical priors selection

Seven sets of lexical priors were defined in an attempt to steer the model towards topics corresponding to key spacecraft subsystems: *AOCS*, *Communication*, *Environment*, *OBDH*, *Power*, *Propulsion*, and *Thermal*. Each set is composed of around 20 words. Each word can only belong to one set to avoid topic overlap. The priors selected, presented in Table 4, are based on a list of keywords or relevant concepts associated with each topic. The list is validated by the same human annotators who performed the manual labelling. A first priors list is distributed to the human annotators who debated which concepts to keep or discard.

Table 4. Set of lexical priors per topics (organised alphabetically)

Topic Label	Lexical Priors
Attitude and Orbit Control Subsystem (AOCS)	angular, attitude, attitude control, body, freedom, gravity gradient, guidance, gyroscope, magnetotorquers, momentum, motion, navigation, reaction wheel, sensor, spin stabilised, stabilisation, star tracker, torque, torquer, wheel
Communication	antenna, band, bandwidth, c-band, command, communication, frequency, ka-band, l-band, receiver, reception, relay, satellite communication, s-band, telecommand, telemetry, tracking, transmitter, packet, x-band,
Environment	background, charging, cosmic, debris, dose, electron, environment, gamma radiation, gamma ray, geomagnetic, particle, protection, radiation, ray, shield, single event, shielding, single event upset, space debris, van allen
On-Board Data Handling	bit, bitrate, computer, cpu, data, data handling, data rate, decoder, downlink, dram, encoder, execution, gbit, instruction, measurement, memory, operation, processor, ram, sram, storage, tag, uplink
Power	battery, battery powered, cell, charge, circuit, current, cycle, depth of discharge, discharge, energy, lithium, photovoltaic, power, power supply, primary, secondary, solar cell, solar power, voltage, watt
Propulsion	delta v, electric, electric propulsion, engine, exhaust, fuel, impulse, ion, isp, nuclear, plasma, propellant, propellant mass, propulsion, propulsion system, sail, spacecraft propulsion, thrust, thruster, total impulse
Thermal	coating, cooling, degree, heat, heat pipe, heater, heating, insulation, louver, mirror, multi layer insulation, radiator, reflective, reflector, temperature, thermal, thermal control, thermal control system, thermodynamics, overheating

5.5. Optimised weighted sum

The WS approach acts on the level of the per-document topic distribution. $\hat{\theta}_i$ denotes the aggregated topic distribution for the unseen data, document i. The WS combines the topic distributions $\theta_{(i,j)}$ of each model M_j for the same document i but balanced by a model weight w_j as shown on equation 3 based on (Wilcox, 2013).

$$\hat{\theta}_i = \frac{\sum_{j=1}^{M} w_j \theta_{(i,j)}}{\sum_{j=1}^{M} w_j} \tag{3}$$

Since this method does not produce new word distributions, it does not entail the tedious process of relabelling. The weights are optimised with a Tree of Parzen Estimators (TPE) algorithm (Bergstra et al., 2011) available in the hyperopt Python library (Bergstra et al., 2013). Fig. 4 summarises the WS aggregation approach.

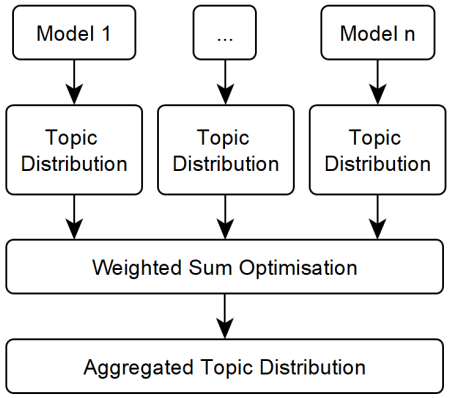

Fig. 4. Schema of the proposed aggregation method

5.6. Topic identification of unseen data

The dictionary of the model is used to map words to their ids. A new corpus document-term matrix is generated based on this dictionary. The topic distribution defined by the spaceLDA model can then be applied to the input

requirement or query. The output is a list of latent topics along with their probability to represent the document.

5.7. Case study evaluation

5.7.1. Accuracy score

The accuracy score only takes into consideration the primary topic of a per-document topic distribution. If this topic matches the requirement's ground truth, then the matching is considered a success. The accuracy score is divided by the number of unseen data, or requirements, submitted to the model. Therefore, the best performance corresponds to an accuracy score of 1.

5.7.2. Mean reciprocal ranking

The MRR takes into consideration the top n topics of a per-document topic distribution. The score is inversely proportional to the correct answer, topic rank, as shown in Equation 4 based on the definition from (Craswell, 2009):

$$MRR = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \frac{1}{rank_i} \tag{4}$$

with Q the number of queries, $rank_i$, the rank of the ground truth. Only the top two topics will be taken into consideration.

6. Results

In this section, we first present the results of the spaceLDA model's hyperparameters optimisation resulting in the selection of topics numbers. We then compare the per-topic word distributions found by the non-aggregated spaceLDA models with distributions generated by unsupervised LDA models trained on the same domain-specific corpora. We finally assess the case study performances of our model while comparing them with the performances achieved by several other methods summarised in Table 5.

Table 5. Overview of compared methods in the case study

	spaceLDA	**method a**	**method b**	**method c**
Training with lexical priors	X	X		
Unsupervised training			X	X
Optimised WS aggregation	X		X	
JS divergence aggregation		X		X

6.1. Hyperparameters optimisation

For each training corpus, the optimisation process described in paragraph 5.1 is run to identify the optimum number of latent topics. The optimisation process was run for a number of topics ranging from 4 to 100 and for each training set. The resulting average perplexity measures are displayed on Fig. 5 up to 40 topics.

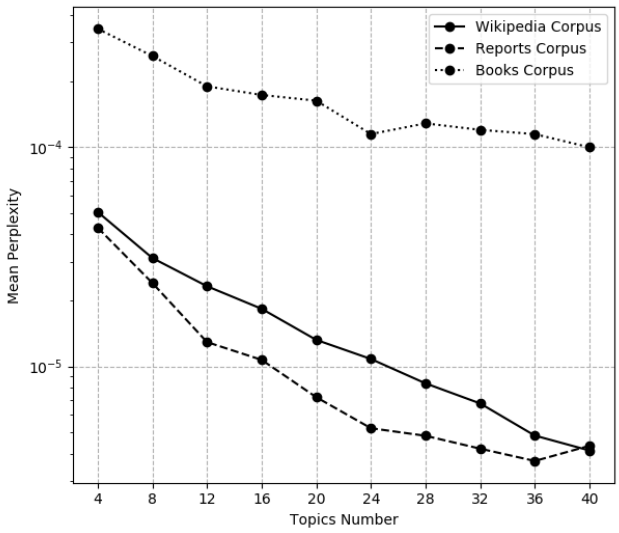

Fig. 5. Perplexity evolution

Setting a perplexity threshold to 0.9E-4 and manually investigating the topics content of models satisfying the perplexity threshold, a topic number of 22

was chosen for the Wikipedia training set. Following the same process, a topic number of 30 latent topics was selected for the reports training set. A model was trained with this set, obtaining a perplexity of 1.35e-06 in its final evaluation, computed with the held-out part of the training corpus. The higher number of latent topics was assumed to result from the variety of missions covered by the reports, as well as the higher complexity of the experts' lexicon writing these reports. For the books training set, a topic number of 24, corresponding to the local minimum, was chosen. A final model obtained a perplexity score of 1.85e-05, similar scores to the above models, although the values of the mean perplexity measures are generally higher for this training set.

6.2. SpaceLDA per-topic word distributions

The LDA model training is a stochastic process; however, the results presented in this subsection represent trends observed with several trained models. The spaceLDA models were trained using the 164 lexical priors presented in Table 4. Not all lexical priors could be found in the corpora' dictionaries. Therefore nineteen words, including the terms 'bit', 'cpu', 'magnetotorquers', 'power supply', 'satellite communication', 'spacecraft propulsion' and 'spin stabilised', were not boosted. Unsupervised models were also trained on the same corpus to compare the latent topics extracted with and without lexical priors.

Tables 6, 7 and 8 display the top 5 terms of per-topic word distributions extracted from the training corpora by the spaceLDA and the unsupervised LDA models. In these tables, the lexical priors have been underlined in bold. Terms found in unsupervised word distributions that happened to match a lexical prior were also underlined for comparison purposes. The complete word distributions of each model are available at `https://github.com/strath-ace/smart-nlp`.

The distributions obtained from the Wikipedia training corpus are similar for both training approaches. With the reports training corpus, complex phrases (more than 2-grams words) such as '*ultra high frequency*' are given less attention in the semi-supervised distributions influenced by lexical priors. The unsupervised distributions promote terms which are less domain-specific,

such as '*laser*' (found in *Environment* topic), '*bipods*'(found in *Thermal* topic) or '*telescope*' (found in *Thermal* topic). Finally, the unsupervised distributions extracted from the books appear to mix different topics. For instance, the *Communication* topic unsupervised word distribution includes the term 'low thrust', a propulsion concept. The *Propulsion* topic distribution includes the term *thermal control*. Overall the word distributions extracted by the spaceLDA model provided a more accurate representation of each spacecraft subsystem. The lexical priors notably enabled to remove noisy terms from distributions based on the reports and books corpora.

Table 6. Comparison of per-topic word distributions obtained from spaceLDA and unsupervised models trained with the Wikipedia corpus. Terms in bold corresponds to lexical priors.

Topic Label	**Training**	**Topic Word Distribution Top 5 elements**
Attitude and Orbit Control System	spaceLDA	**momentum**, **angular**, velocity, **motion**, particle
	Unsupervised	**attitude**, **sensor**, **wheel**, orientation, **momentum**
Communication	spaceLDA	radio, **frequency**, signal, **antenna**, **receiver**
	Unsupervised	radio, **antenna**, **receiver**, signal, wave
Environment	spaceLDA	**cosmic**, **radiation**, **particle**, belt, allen
	Unsupervised	**radiation**, gamma, **cosmic**, **particle**, decay
On-board Data Handling	spaceLDA	**memory**, dynamic random access memory, cable, **data**, cell
	Unsupervised	**memory**, dynamic random z access memory, cable, cell, **computer**
Power	spaceLDA	**cell**, **power**, **photovoltaic**, pressurized pressure vessel, electricity
	Unsupervised	capacitor, **voltage**, **circuit**, capacitance, resistance
Thermal	spaceLDA	**heat**, **heating**, **temperature**, material, **thermal**
	Unsupervised	**heat**, **temperature**, **thermal**, **heat pipe**, **cooling**

Table 7. Comparison of per-topic word distributions obtained from spaceLDA and unsupervised models trained with the reports corpus. Terms in bold corresponds to lexical priors.

Topic Label	**Training**	**Topic Word Distribution Top 5 elements**
Communication	spaceLDA	**x-band**, **band**, **telemetry**, modulation, **telecommand**
	Unsupervised	ultra high frequency, orbiter, localisation, conjunction, arrival
Environment	spaceLDA	**radiation**, **shielding**, **particle**, **environment**, **electron**
	Unsupervised	bench, interferometer, decoherence, nanoparticles, laser
Power	spaceLDA	**energy**, capacity, panel, solar power, **voltage**
	Unsupervised	laser interferometer space antenna, electric propulsion, laser, constellation, telescope
Propulsion	spaceLDA	**propellant**, transfer, refuelling, optical, geostationary orbit
	Unsupervised	**electric propulsion**, asteroid, eprop, boost, arrival
Thermal	spaceLDA	**overheating**, **thermodynamics**, **reflective**, **thermal control system**, **thermal control**
	Unsupervised	**cooling**, telescope, cryogenic, spectro, bipods

Table 8. Comparison of per-topic word distributions obtained from spaceLDA and unsupervised models trained with the books corpus. Terms in bold corresponds to lexical priors.

Topic Label	**Training**	**Topic Word Distribution Top 5 elements**
Attitude and Orbit Control Subsystem	spaceLDA	**attitude**, vector, matrix, control, frame
	Unsupervised	quaternion, covariance, kalman, kinematics, architect
Communication	spaceLDA	orbit, service, data, **antenna**, **communication**
	Unsupervised	decentralized, nonlinear, synchronization, topology, low thrust
On-Board Data Handling	spaceLDA	on board computer, **data**, interface, frame, power, channel
	Unsupervised	on board computer, power control and distribution unit, connector, register, spacewire
Propulsion	spaceLDA	**propulsion**, **fuel**, electric, **thruster**, **propulsion system**
	Unsupervised	nozzle, packet, combustion, thermal control, qualification
Thermal	spaceLDA	**thermal**, **heat**, **temperature**, control, orbit
	Unsupervised	**thermal control**, **insulation**, **coating**, pumped, vapour

Fig. 6 displays the labels of extracted distributions for each corpus. Within the Wikipedia corpus, most of the topics of interest could be identified. The variation between the spaceLDA models and the unsupervised training is again minimal for this corpus. The benefits of lexical priors become more apparent for the remaining training sets. The diversity of topics extracted increased in both cases with the introduction of lexical priors. The priors encourage the extraction of topics that might be less prevalent and are then overlooked by the unsupervised model. In the case of the reports corpus, it is clear that the

unsupervised model focused on two topics, *Propulsion* and *Environment*. The priors balanced this attention, enabling the expression of other topics of interest. Similarly, in the case of the books corpus, several noisy topics (tagged as *Other* and not represented in the Figure) were extracted by the unsupervised model. In conclusion, the lexical priors contributed to the extraction of more relevant topics.

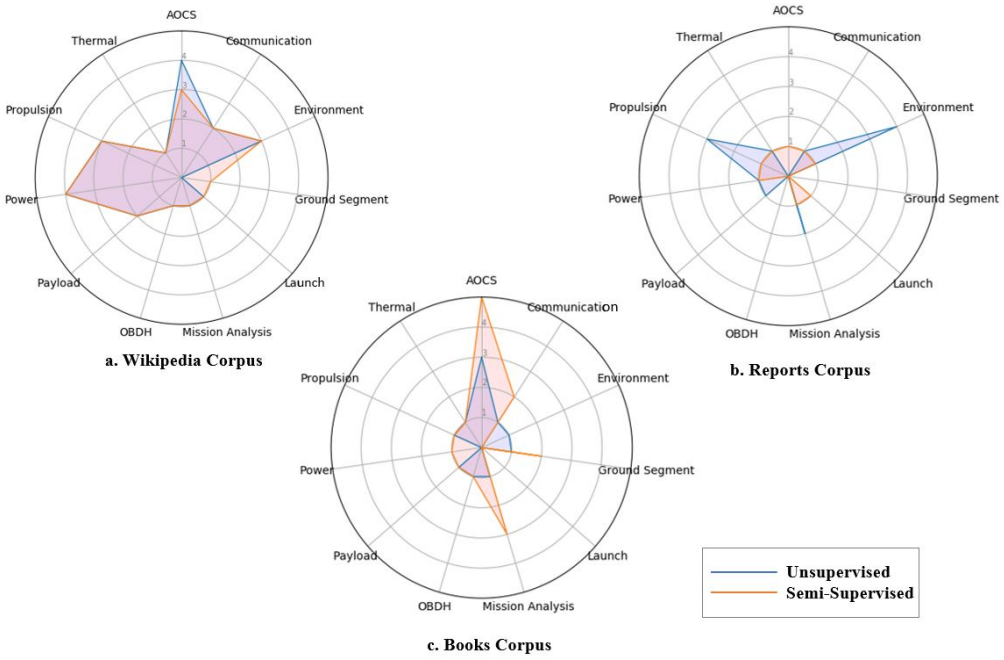

Fig. 6. Evolution of topics labelling between unsupervised models (blue distribution) and spaceLDA (semi-supervised) (orange distribution) for each training set.

The averages of lexical priors found in the top 50 of word distributions are displayed in Fig. 7. For the unsupervised models, the priors are not used as lexical priors as they don't interfere in the training process, however, they can still appear in the word distributions. For the model trained with the Wikipedia set, the unsupervised topics, considering all categories, contain on average 18,1% of lexical priors against 19,2% for the spaceLDA model. For the model trained with the reports or books sets, the difference is far more prominent. For the reports training set, over all categories, the average for the spaceLDA topics is 44%, compared to 4% for the unsupervised model. Similarly, for the books corpus, the average number of lexical priors increased from 8% to 31% with the spaceLDA approach.

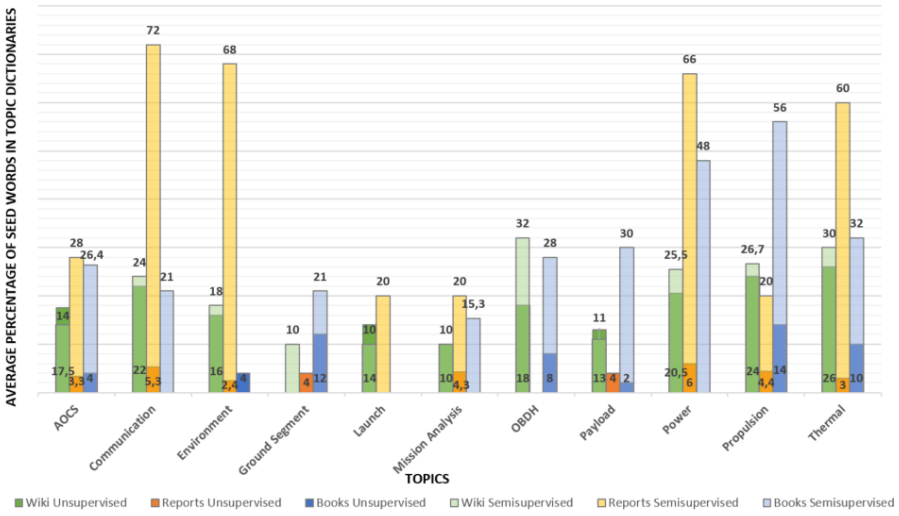

Fig. 7. Average percentage of lexical priors found in each topic distribution and each training set with the spaceLDA (semi-supervised) or unsupervised models.

6.3. Case study results

In this case study, design requirements are submitted as unseen documents to the spaceLDA model and to three other models introduced in Table 5. The requirement's prevalent topic found by the TM models is then compared to the requirement's ground truth. To ensure robustness, the training process is run ten times for both training methods and for each training set, yielding in total 30 spaceLDA models and 30 unsupervised LDA models. Each aggregation method is applied to each set of 30 models.

6.3.1. Aggregation of per-document topic distribution with an optimised weighted sum

With our aggregation method, the merging occurs after each model has generated a topic distribution for the unseen document. The contribution of each model is balanced with weights to optimise the categorisation performance. The hyperparameter optimisation ran with the hyperopt Python library yields the following linearised weight combinations:

1. SpaceLDA models: 0.54 for Wikipedia-based models, 0.05 for feasibility reports-based models, and 0.41 for book-based models.
2. Unsupervised models: 0.62 for Wikipedia-based models, 0.19 for feasibility reports-based models, and 0.19 for book-based models.

For this case study, the weights of Wikipedia-based models are significantly higher than for other training corpora. Indeed, selected Wikipedia pages are more likely to describe the architecture of space systems than the feasibility reports detailing applied examples of spacecraft design or books covering broader topics. Therefore, the Wikipedia corpus is given more influence with a higher weight. The weights would need to be fine-tuned again should the case study change.

6.3.2. *Aggregation of per-topic word distribution with the Jensen-Shannon divergence*

As presented in (Blair et al., 2020), the JS divergence enables the symmetric measurement of similarity between two or more probability distributions. A value of the JS divergence equal to 0 indicates a complete similarity and a value of 1 a complete dissimilarity. Provided the divergence between n similar topics of M models with T_i topics is lower than the JS divergence threshold, γ, an aggregated topic $\hat{\varphi}_k$ can be generated following equation 5 based on (Blair et al., 2020). $\varphi_{(i,j)}$ being the per-topic word distribution of topic T_j in model M_i.

$$\hat{\varphi}_k = \begin{cases} \sum_{i=1}^{M} \sum_{j=1}^{T_i} \frac{\varphi_{(i,j)}}{n}, & \text{if } D_{JS}(\varphi_{(i,j)} || \varphi_x) \leq \gamma \\ 0, & \text{otherwise} \end{cases} \tag{5}$$

The aggregation process is separately run on the 30 unsupervised models and on the 30 spaceLDA models. In each case, 30 models (10 per training set) amount to 780-word distributions to be aggregated into one model. To compare the distributions obtained from heterogeneous sources, a new dictionary is generated based on the vocabulary gathered from all word distributions. The probability distributions are reorganised according to this common dictionary. The JS divergence is computed for each per-topic word distribution with regards to the 779 other distributions. The average JS divergence for the semi-supervised models is 0.68, and for the unsupervised models, 0.80. The semi-supervised models tend to focus on more similar topics due to the lexical priors. A threshold of 0.3 is set to retain only the closest distributions. All topic distributions with a JS divergence lower than this threshold are aggregated. Otherwise, topics are kept as such. Eventually, the unsupervised models yield one aggregated model with 559 topics, while the semi-supervised models yield one aggregated model with 505 topics. The topics are manually labelled by human annotators. The unseen data will be converted into a bag of words based on the aggregated model dictionary.

6.3.3. Categorisation results and comparison

The unseen data whose topics will be identified are 68 mission requirements, as presented in 4.3, related to the topics of *AOCS*, *Communication*, *Environment*, *OBDH*, *Power*, *Propulsion*, and *Thermal*. The documents' chapters from which the requirements are extracted are used as ground truths. Perplexity and Mean Reciprocal Ranking are used to evaluate the models' performances. The performances of the different models are compared in Tables 9-10.

Table 9. Categorisation Accuracy - the highest score per category are underlined in bold and the results of the proposed method are highlighted in the grey column.

Training		Lexical Priors		Unsupervised	
Aggregation Method		WS	JS Divergence	WS	JS Divergence
Labels	**AOCS**	**0.64**	0.54	**0.64**	0.36
	Communication	**0.7**	0.3	0.6	0.2
	Environment	**0.2**	**0.2**	**0.2**	0.1
	OBDH	**0.6**	0.2	0.3	0.1
	Power	0.64	0.2	**0.73**	0.36
	Propulsion	**0.8**	0.4	**0.8**	0.2
	Thermal	**0.73**	0.09	0.36	0.64
Accuracy of aggregated models		**0.61**	0.28	0.52	0.28

Table 10. Categorisation MRR - the highest score per category are underlined in bold and the results of the proposed method are highlighted in the grey column.

Training		Lexical Priors		Unsupervised	
Aggregation Method		WS	JS Divergence	WS	JS Divergence
Labels	**AOCS**	**0.73**	0.59	**0.73**	0.5
	Communication	**0.75**	0.4	0.65	0.2
	Environment	0.2	**0.3**	0.25	0.15
	OBDH	**0.7**	0.35	0.3	0.2
	Power	0.68	0.3	**0.77**	0.55
	Propulsion	**0.8**	0.5	**0.8**	0.4
	Thermal	**0.82**	0.32	0.55	**0.82**
MRR of aggregated models		**0.67**	0.39	0.58	0.40

In Tables 11-12 samples of per-requirement topic distributions obtained with both training and aggregation approaches are displayed. Each topic distribution is a probabilistic distribution of the topics most likely to be found in the assessed requirement. In the case of the propulsion requirement, Table 11, all methods successfully associate the requirement to its correct category either as a first or second most prevalent topic. Although this requirement was assigned to the propulsion subsystem chapter in (ESA, 2005), the requirement clearly incorporates notions of thermal management. The unsupervised model aggregated with the JS divergence is thus led to confusion by the terms "thermal design" and "temperature". This duality is also reflected by the distributions obtained with the weighted sum, which indicate that *Thermal* is also a prevalent topic of the requirement.

Table 11. Example of topic distributions obtained for a propulsion requirement. The ground truth topic is underlined in bold in the distributions. The *other* label stands for a topic unrelated to space subsystems.

Propulsion Requirement		The propulsion sub-system thermal design shall assure that the minimum predicted temperatures of any wetted component or surface contacting the propellant remain at least 10oC above the maximum freezing point of the onboard propellant.
Training	**Aggregation**	**Topic Distribution**
Lexical Priors	WS	**'propulsion'**: 0.35, 'thermal': 0.25
	JS	**'propulsion'**: 0.43, 'other': 0.30
Unsupervised	WS	**'propulsion'**: 0.43, 'thermal': 0.22
	JS	'thermal': 0.52, **'propulsion'**: 0.27

As seen in Tables 9-10, the categorisation of *Power* requirements were the only classes for which the performances of the spaceLDA models were lower than for the other approaches. However, even then, the WS approach, relying on the unsupervised models' aggregation, performed better than the JS divergence aggregation. Therefore, to improve the performance of the spaceLDA model, the lexical priors used to identify and define the *Power* topics should be improved. In Table 12 for instance, the WS method is the only one to properly associate the first topic to the requirement to its ground truth.

Table 12. Example of topic distributions obtained for a power requirement. The ground truth topic is underlined in bold in the distributions. The *other* label stands for a topic unrelated to spacecraft subsystems.

Power Requirement		Cell performance and degradation factors shall be justified according to in orbit experience and supporting ground testing.
Training	**Aggregation**	**Topic Distribution**
Lexical Priors	WS	'other': 0.24, **'power'**: 0.17
	JS	'other': 0.92
Unsupervised	WS	**'power'**: 0.27, 'propulsion': 0.16
	JS	'other': 0.54, **'power'**: 0.26

7. Discussion

The training of a domain-specific LDA model required a curated domain-specific corpus. The documents integrated into the training data set were representative of the texts engineers usually rely on to master space systems. The heterogeneity of the training data set had the advantage of diversifying the topics discovered. To ensure that the larger data set would not overshadow the smaller corpora, the models were trained independently on three training subsets. As it could be expected, the semi-supervised training of the spaceLDA model yielded better results than the classic LDA unsupervised training. The outputs of the unsupervised models were, however, useful in supporting the definition of the lexical priors which were used for the spaceLDA.

The heterogeneity of the training data set also meant that for each unseen document, corresponding to a design requirement, as many topic distributions as models were found. Hence our proposition to merge distributions with an optimised WS to converge towards a single per-requirement topic distribution, enabling us to match each requirement with one key spacecraft subsystem even-

tually. Our aggregation method outperformed the JS divergence aggregation method. The WS was a more flexible option allowing to balance the influence of the different corpora, enabling fine-tuning. In addition, this method did not require relabeling the topics of the merged model, enabling quicker reuse of pre-trained models. The WS optimisation assigned heavier weights to the Wikipedia corpus. This was expected as the per-topic word distributions based on the Wikipedia corpus covered most of the topics of interest and were of similar quality for the spaceLDA and unsupervised training. The selected Wikipedia pages were more likely to efficiently and shortly describe the architecture of space systems, while the feasibility reports provided applied examples of spacecraft design and books covered broader topics.

8. Conclusion and future work

This study introduced a novel domain-specific TM model, spaceLDA, tailored to space systems, enriched with lexical priors and an optimised WS. The development of the spaceLDA model entailed the generation and curation of a first text collection on space systems as well as the definition of domain-specific lexical priors. The practical application of TM to support space mission design was demonstrated through a case study on the categorisation of design requirements. The study proposed an optimised WS to aggregate per-document topic distributions. This method outperformed the state of the art aggregation method based on the JS divergence. The models and corpora, with the exception of the feasibility reports, are available at `https://github.com/strath-ace/smart-nlp`. Although applied to a corpus related to space systems, the approach proposed here is extendable to any domain-specific corpus. In future work, the application of TM could be extended to the classification of documents. The TM approach could be compared to a word embedding approach. Labels could be automatically assigned to topics to mitigate the subjectivity of the manual labelling.

Acknowledgements

This study was completed in the frame of the Design Engineering Assistant (DEA) project, a virtual assistant to support knowledge management, reuse, and decision-making at the early stages of space mission design. The DEA is developed in the frame of an ESA Networking Partnership Initiative (NPI), the authors would like to warmly thank their partners for their valuable support: ESA, RHEA Systems, AIRBUS, and satsearch.

References

B. Al-Salemi, M. Ayob, S. A. M. Noah, and M. J. A. Aziz. Feature selection based on supervised topic modeling for boosting-based multi-label text categorization. *Proceedings of the 2017 6th International Conference on Electrical Engineering and Informatics (ICEEI)*, pages 1–6, 2017. doi: 10.1109/ICEEI.2017.8312411.

L. Alsumait, D. Barbara, and C. Domeniconi. On-Line LDA : Adaptive Topic Models for Mining Text Streams with Applications to Topic Detection and Tracking. In *8th IEEE International Conference on Data Mining*. IEEE, 2008. doi: 10.1109/ICDM.2008.140.

R. Bahmanyar, D. Espinoza-Molina, and M. Datcu. Multisensor Earth Observation Image Classification Based on a Multimodal Latent Dirichlet Allocation Model. *IEEE Geoscience and Remote Sensing Letters*, 15(3):459–463, 2018. ISSN 15580571. doi: 10.1109/LGRS.2018.2794511.

J. Bergstra, R. Bardenet, Y. Bengio, and B. Kégl. Algorithms for Hyper-Parameter Optimization. In *NIPS'11: Proceedings of the 24th International Conference on Neural Information Processing Systems*, 2011.

J. Bergstra, D. Yamins, and D. D. Cox. Making a Science of Model Search : Hyperparameter Optimization in Hundreds of Dimensions for Vision Architectures. In *30th International Conference on Machine Learning*, Atlanta, Georgia, USA, 2013.

A. Berquand, F. Murdaca, A. Riccardi, T. Soares, S. Generé, N. Brauer, and K. Kumar. Artificial Intelligence for the Early Design Phases of Space Missions Artificial Intelligence for the Early Design Phases of Space Missions. In *IEEE Aerospace*, number March, 2019. ISBN 9781538668542. doi: 10.1109/AERO.2019.8742082.

S. Bird, E. Klein, and E. Loper. *Natural Language Processin with Python*. O'Reilly Media Inc., 2009. ISBN 9780596516499.

G. C. Birur, G. Siebes, and T. D. Swanson. Spacecraft Thermal Control. In *Encyclopedia of Physical Science and Technology*, pages 485–505. Elsevier, 2003. ISBN 9780857091567. doi: 10.1016/B0-12-227410-5/00900-5. URL `https://linkinghub.elsevier.com/retrieve/pii/B0122274105009005`.

S. J. Blair, Y. Bi, and M. D. Mulvenna. Aggregated topic models for increasing social media topic coherence. *Journal of Applied Intelligence*, 50:138–156, 2020. doi: https://doi.org/10.1007/s10489-019-01438-z.

D. M. Blei and J. D. Lafferty. A Correlated Topic Model of Science. *The Annals of Applied Statistics*, 1(1):17–35, 2007. doi: 10.1214/07-AOAS114.

D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent Dirichlet Allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003. URL `http://www.jmlr.org/papers/volume3/blei03a/blei03a.pdf`.

L. Chen, H. Zhang, J. M. Jose, H. Yu, Y. Moshfeghi, and P. Triantafillou. Topic detection and tracking on heterogeneous information. *Journal of Intelligent Information Systems*, 51(1):115–137, 2018. ISSN 15737675. doi: 10.1007/s10844-017-0487-y.

N. Craswell. Mean Reciprocal Rank. In *LIU L., ÖZSU M.T. (eds) Encyclopedia of Database Systems*. Springer, Boston, MA, 2009. doi: https://doi.org/10.1007/978-0-387-39940-9{_}488.

ECSS. ECSS Terms and Definitions. Technical report, 2007. URL `https://ecss.nl/home/ecss-glossary-terms/`.

ECSS. ECSS Abbreviated Terms. Technical report, 2017. URL https://ecss.nl/home/ecss-glossary-abbreviations/.

ESA. SMOS Systems Requirements Document. Technical report, 2005. URL https://sci.esa.int/web/marcopolo-r/-/51297-marcopolo-r-mission-requirements-document/.

ESA. MarcoPolo-R Mission Requirements Document. Technical report, 2012. URL https://sci.esa.int/web/marcopolo-r/-/51297-marcopolo-r-mission-requirements-document/.

T. Iqbal, P. Elahidoost, and L. Lúcio. A Bird's Eye View on Requirements Engineering and Machine Learning. In *2018 25th Asia-Pacific Software Engineering Conference (APSEC)*, pages 11–20, Nara, Japan, 2018. ISBN 9781728119700. doi: 10.1109/APSEC.2018.00015.

J. Jagarlamudi, H. Daumé III, and R. Udupa. Incorporating Lexical Priors into Topic Models. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 204–213, Avignon, France, 2012. Association for Computational Linguistics.

S. J. Kapurch. NASA Systems Engineering Handbook. *NASA Special Publication*, page 360, 2007. URL http://adsabs.harvard.edu/full/1995NASSP6105.....S.

S. Karasu and A. Altan. Recognition Model for Solar Radiation Time Series based on Random Forest with Feature Selection Approach. In *11th International Conference on Electrical and Electronics Engineering (ELECO)*, pages 8–11, Bursa, Turkey, 2019. doi: 10.23919/ELECO47770.2019.8990664.

W. J. Larson and J. R. Wertz. *Space Mission Analysis and Design*. Third edition, 2005. ISBN 9780792359012. URL http://www.amazon.ca/Space-Mission-Analysis-Design-James/dp/0792359011/ref=sr_1_1?s=books&ie=UTF8&qid=1430653992&sr=1-1&keywords=9780792359012.

L. Layman, A. P. Nikora, J. Meek, and T. Menzies. Topic Modeling of NASA Space System Problem Reports Research in Practice. In *IEEE/ACM 13th Working Conference on Mining Software Repositories Topic*, 2016. ISBN 9781450341868.

M. Liénou, H. Maître, and M. Datcu. Semantic Annotation of Satellite Images Using Latent Dirichlet Allocation. *IEEE Geoscience and Remote Sensing Letters*, 7(1):28–32, 2010. doi: 10.1109/LGRS.2009.2023536.

F. Liu, S. Lu, and Y. Sun. *Guidance and Control Technology of Spacecraft on Elliptical Orbit*. 2019. ISBN 9789811079580.

T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient Estimation of Word Representations in Vector Space. In *Proceedings of the International Conference on Learning Representations (ICLR 2013)*, 2013. URL `http://arxiv.org/abs/1301.3781`.

Y. Moshfeghi, B. Piwowarski, and J. M. Jose. Handling Data Sparsity in Collaborative Filtering using Emotion and Semantic Based Features. In *34th International ACM SIGIR Conference on Research and Development in Information*, pages 625–634, Beijing, China, 2011. ISBN 9781450307574. doi: 10.1145/2009916.2010001.

J. S. Park, N. R. Kim, H. R. Choi, and E. Han. A new forecasting system using the latent dirichlet allocation (LDA) topic modeling technique. *WSEAS Transactions on Environment and Development*, 14:363–373, 2018. ISSN 22243496.

R. Řehůřek and P. Sojka. Software Framework for Topic Modelling with Large Corpora. (May 2010), 2014. doi: 10.13140/2.1.2393.1847.

M. Rosen-Zvi, T. Griffiths, M. Steyvers, and P. Smyth. The Author-Topic Model for Authors and Documents. In *UAI '04: Proceedings of the 20th conference on Uncertainty in artificial intelligence*, pages 487–494, 2004.

C. Schnober and I. Gurevych. Combining Topic Models for Corpus Exploration
640 Applying LDA for Complex Corpus Research Tasks in a Digital Humanities Project. In *TM '15: Proceedings of the 2015 Workshop on Topic Models: Post-Processing and Applications*, Melbourne, Australia, 2015. ISBN 9781450337847. doi: http://dx.doi.org/10.1145/2809936.2809939.

A. P. Shiryaev, A. V. Dorofeev, A. R. Fedorov, L. G. Gagarina, and V. V.
645 Zaycev. LDA Models for Finding Trends in Technical Knowledge Domain. In *2017 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus)*, pages 551–554. IEEE, 2017. doi: 10.1109/EIConRus.2017.7910614.

W. Sriurai. Improving Text Categorization By Using A Topic Model. *Advanced Computing: An International Journal (ACIJ*, 2(6):21–27, 2011. ISSN
650 2229726X. doi: 10.5121/acij.2011.2603.

The Apache Software Foundation. Apache Tika 1.20, 2018. URL `https://tika.apache.org/1.20/index.html`.

C. Văduva, C. Dănişor, and M. Datcu. Joint SAR image time series and PSIn-
655 SAR data analytics: An LDA based approach. *Remote Sensing*, 10(9), 2018. ISSN 20724292. doi: 10.3390/rs10091436.

R. Wilcox. Chapter 6 - Some Multivariate Methods. In *Introduction to Robust Estimation and Hypothesis Testing*, number 3rd, pages 215–289. 2013. ISBN 9780123869838. doi: 10.1016/b978-0-12-386983-8.00006-8.

660 J. Yun and Y. Geum. Automated classification of patents: A topic modeling approach. *Computers and Industrial Engineering*, 147(July):106636, 2020. ISSN 03608352. doi: 10.1016/j.cie.2020.106636. URL `https://doi.org/10.1016/j.cie.2020.106636`.