47 International Journal of Technology v(i) pp-pp (YYYY)
Received Month Year / Revised Month Year / Accepted Month Year

International Journal of Technology

http://ijtech.eng.ui.ac.id

Clustering Narrow-Domain Scientific Text using Unsupervised and Similarity-based Approach

Put authors name here[1i], Put author name here[1], Put author name here[1], Put author name here[2], Put authors name here[2]

[1]*Put author's affiliation here, complete with address, postal code, and country*
[2]*Put author's affiliation here, complete with address, postal code, and country*

Note: Due to our double-blind review policy, please include authors' information (including in the header and footer) only after acceptance at the peer review process.

Abstract. Clustering scientific papers published by the authors is useful for discovering fellow authors with similar interests or research groups in the institution. In this paper, we explore the use of scientific text clustering with an unsupervised approach to enhance the efficiency of retrieving similar works. Challenges in clustering scientific papers from a specific domain include an increase in the list of non-discriminating words (stop words) because there are more words that are becoming common in most of the documents. For example, words like *engineering* will no longer have discriminating power if most documents come from the engineering field. There is also a challenge from the use of similar terminologies to express different concepts, such as *internet* vs. *internet of things*. To address this, we experimented with various text processing methods, including stemming, lemmatization, technical stop word removal, noun extraction, and n-gram phrase detection. The experiment was conducted on a corpus of publications from our faculty. Our methodology used the text processing methods with Latent Dirichlet Allocation (LDA) and Non-Negative Matrix Factorization (NMF) topic models to cluster the documents and uncover latent topics within the corpus. The optimal clustering pipeline was determined to be the NMF model combined with lemmatization, technical stop word removal, noun extraction, and phrase detection. The pipeline yielded eleven clusters with the following evaluation scores: UMass of -2.493, CV of 0.681, NPMI of -0.136, and UCI of -4.491. It has also improved a sample accuracy from 71.1% to 80.7% and generalized well to a different dataset. The resulting clusters from this pipeline fit our institution's research groups, such as electrical power engineering, signal processing, and computer vision. Additionally, we provide the curated list of technical stop words that contributed to the effectiveness of our clustering results.

Keywords: Narrow-Domain Text Clustering; Latent Dirichlet Allocation; Non-Negative Factorization Matrix; Text Processing; Topic modelling

1. Introduction

The increasing number of scientific publications provides more research ideas and acts as references for future innovations to come (Larsen & von Ins, 2010; Li et al., 2020). However, it also raises the need for an efficient search process in handling a vast repository of texts. This is useful for upcoming authors to explore supporting works of literature for their future research as well as to discover fellow authors with similar interests (Sajid et al., 2021). To increase the

[i] Corresponding author's email: name@ai.ue.oa, Tel.: +00-00-000000; Fax: +00-00-000000
doi: 10.14716/ijtech.v0i0.0000

quality of the search process, document clustering is one of the approaches that could be used to label a publication into the most suitable groups. The clustering results support retrieval processes to return more relevant documents (Kadhim, 2019; Zibani et al., 2022). Advances in text analysis tasks, such as text classification (Aftab et al., 2023; M. Mohammed et al., 2021; Tey et al., 2023) and document grading (Lubis et al., 2021), are also reflected on text clustering (M. Mohammed et al., 2021), thus enabling more possible cases to be explored.

Moreover, document clustering delivers insight into documents within a collection, especially on conditions when documents are unannotated. Within an academic institution, even in one with established research groups, clustering research documents can help to identify emerging research topics. Discovering topic groups enables seeing the occurring research trends which can help determine the research direction an institution is heading towards. Clustering can automate managing the scientific publications archive as it can be used to assist labelling documents automatically.

There have been notable approaches to clustering scientific documents in an institution (Bellaouar et al., 2021; Kim & Gil, 2019; Pavithra & Savitha, 2024; Preetham M C et al., 2022). The work presented in those papers aimed to discover groups of research interest existing within the faculty. Insights discovered from the clustering process helps members within the faculty to learn about the research focus in the faculty. Furthermore, this insight gives references to the topics open for further developments, thus igniting collaboration within faculty members.

Document clustering is a technique to group documents into clusters where documents in a cluster share common properties according to defined similarity measures (Shah & Mahajan, 2012). In contrast to document classification where the number of clusters and the cluster for each the document is known, document clustering does not have the information of the number, the characteristics, or the members of the clusters. This makes document clustering a type of an unsupervised learning.

Topic modelling is an approach to cluster documents and discover useful topics from each cluster (Muchene & Safari, 2021; Vayansky & Kumar, 2020). Recently, word embeddings could be used to vectorize document contents, then the documents are grouped based on their similarity in the vector space (Mehta et al., 2021). Other approaches to discover insights from text documents includes co-word analysis (Leung et al., 2017; Surjandari et al., 2015). Among those methods, the commonly used approach for topic modelling tasks are Latent Dirichlet Allocation (LDA) and Non-Negative Matrix Factorization (NMF) (Mifrah, 2020; Smail et al., 2023; Yu & Xiang, 2023).

Latent Dirichlet Allocation (Blei et al., 2003) is a generative model which identifies topics from a collection of text documents. This model assumes a document is a mixture of topics, and a topic could be seen as a distribution of words. The LDA learning process aims to discover the distribution of words within a topic. This is achieved by calculating the probability of a word given the topic and updating the probability distribution for each topic. The graphical model of LDA to generate k topics could be seen on **Fig 1**, with M as the number of documents and N as the size of vocabulary. LDA have been used in various ways, such as topic modelling in linguistic science, political science, biomedical fields, geographical locations, and social networks (Jelodar et al., 2019). In context of scientific document clustering, it has been used in classifying documents in environmental education (Chang et al., 2021) and profile publications related to industry 4.0 (Janmaijaya et al., 2021).

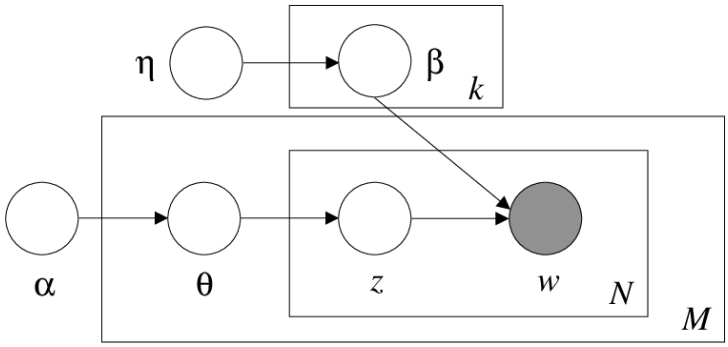

Fig 1. Graphical model of LDA (Blei et al., 2003)

Another topic modelling approach, Non-Negative Matrix Factorization (NMF) (Lee & Seung, 1999) utilizes a nonnegative matrix structure of dimension m x n as the document-term matrix. In this method, a matrix V is approximated as a product of two matrices W and H with the dimensions of m x k and k x n. In topic modelling, k represents the number of topics being discovered from the corpus. Thus, the weights on each vector column on W represent the rank of words within the topic, and the weight on each column of H represent the proportions of topics within that document. NMF is typically used for dimensionally reduction and clustering (Hassani et al., 2021; Tsuge et al., 2001; Wang & Zhang, 2013), and several works have specifically utilized NMF to cluster documents (Laxmi Lydia et al., 2020; Shahnaz et al., 2006). Between the two topic modelling methods, LDA and NMF perform differently on short documents, with the latter considered better at discovering distinct topics (Egger and Yu, 2022).

In this study, we applied topic modelling techniques to discover research interest groups in the dataset of publications from School of Electrical Engineering and Informatics (STEI) Institut Teknologi Bandung. The nature of STEI publications which focused on the field of computer science and electrical engineering raised the difficulty of clustering the documents due to an increase of noise and very closely similar texts.

There is an increase of noise because there are more words that appeared in most, if not all, of the documents. For a word to have a discriminating value, it has to be different. It must appear uniquely on few documents which share some similarity. A word that appears in most documents cannot become an identifying feature to determine the document cluster, thus it become a noise to the dataset. This kind of words are called *stop words.*

The challenge of increased number of stop words in domain specific texts was addressed in previous research (Sarica & Luo, 2021) by identifying domain specific stop words. The research was focused a broader range of engineering. In this paper, we focused on more specific fields with hope to produce more effective collection of technical stop words, especially in the domain of computer science and electrical engineering.

Having texts from very specific domain of engineering also caused the texts to be very similar to each other because similar terminologies were used to explain different concepts. For example, the word *model* in context of machine learning refers to an algorithm trained on data (e.g., neural network model), while *model* in context of software design refers to an abstract representation of a system (e.g., UML model). This ambiguity was addressed by grouping words that describe a concept into phrases, with hope that phrase can better capture the nuance of what was being discussed.

We experimented with several text processing methods before proceeding to document clustering, including the process to removing domain specific stop words and grouping words to phrases, to find out the most optimal pipeline for scientific text clustering. Each experiment

was evaluated with coherence scores as quality metrics, as done in existing works (Hadiat, 2022; Mifrah, 2020). In the end, we also observed and evaluated a sample of our results to gain more understanding on the clusters formed.

The contributions of this paper are as follows:

1. Curated a list of computer science & electrical engineering stop words. We analysed our corpus, which consists of titles and abstracts of computer science and electrical engineering publications, to identify words with no discriminative value (stop words) in computer science and electrical engineering domain.
2. Experimentation to combine various text preprocessing steps. We compared various combinations of text preprocessing steps for discovering topic clusters within scientific publications. The preprocessing methods we experimented include stemming and lemmatization; technical stop words removal; noun extraction; and n-gram phrase detection.
3. Lastly, we proposed an optimum text clustering pipeline that is most suited to our text clustering task based on our observation of our experiment results.

The paper consists of several sections as follows. The first section describes the background for these experiments, which also includes a quick summary of previous works which support our topic. Afterward, the second section provides description of the case study dataset along with our proposed design of the experiment. The experiment results are displayed and analysed further in the following sections. Finally, the last section contains all cited in the manuscript and presents all additional discoveries from the experiments.

2. Methods

2.1. Data

In this research, we used publications from School of Electrical Engineering and Informatics (STEI), Institut Teknologi Bandung as our dataset for this case study. This dataset contains the title, authors, publication year, and abstract of scientific publications authored by the institution's lecturers. This dataset is compiled from STEI owned dataset and various digital publications platforms, which include Google Scholar, and IEEE Xplore. The data were collected from November until December 2022 with a total of 10246 titles.

It is to be noted that after preliminary data cleaning, there were 10246 unique publication titles available. However, determining and removing duplicates present some challenges. One example is among publications that determined to be duplicated, some possess identical titles but are published in different venues and thus count as two separate publications. As such, manual or semi-manual checking using a script may be required for further data cleaning. Citing this complexity, for the sake of this manuscript, analysis is done without removing presumed duplicates.

We obtained the publication year for most (8777 out of 10246) of the publications collected. As shown in **Fig 2,** most publications collected were published after the start of 2006, which coincides with the formation of STEI. Publications dated back before 2006 were mostly books authored by senior lecturers.

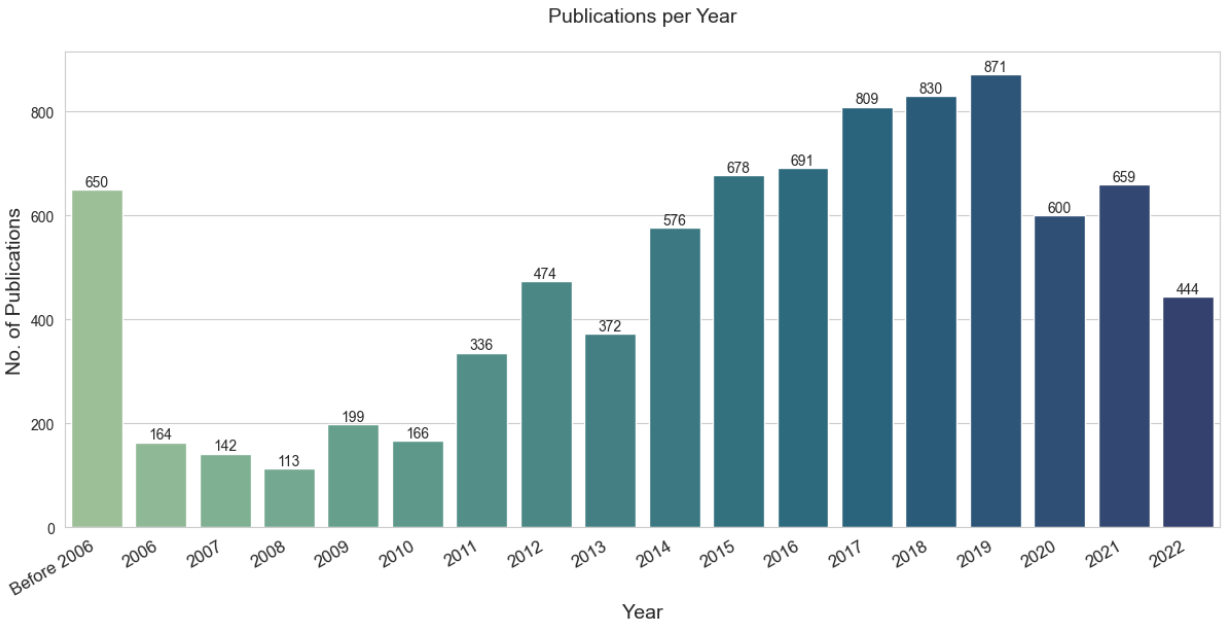

Fig 2. Publication Year Distribution

Most of the collected publications are in English, with a smaller portion in Indonesian and other languages such as Japanese and German. **Table 1** described the distribution of languages used in the publications.

Table 1. Language Distribution of Paper Collection

Language	Number of Paper
English	9330
Indonesian	905
Japanese	3
German	3
Others	5

Each lecturer in the dataset belonged to a specific research group within the faculty, as seen on **Table 2**. Among the 10246 collected, we notice the participating research groups for 9767 publications: the remaining 479 being blank, most likely due to failure in disambiguating the author's name because of different name spelling. We also discovered that there were some publications which were part of more than one research group. For example, a publication on Internet of Things (IoT) is part of the collaboration of lecturers from the EL and IF research groups. By counting the number of entries containing the code for each research group, we obtained the number of publications in which each research group contributed, as shown in **Fig 3**.

Table 2. List of Research Groups

Code	Research Group	Code	Research Group	Code	Research Group
ET	Telecommunication Engineering	EL	Electrical Engineering	EP	Electrical Power Engineering
IF	Informatics	KSE	Knowledge and Software Engineering	EB	Biomedical Engineering

Code	Research Group	Code	Research Group	Code	Research Group
TI	Information Technology	CE	Computer Engineering	SKK	Control and Computer Systems

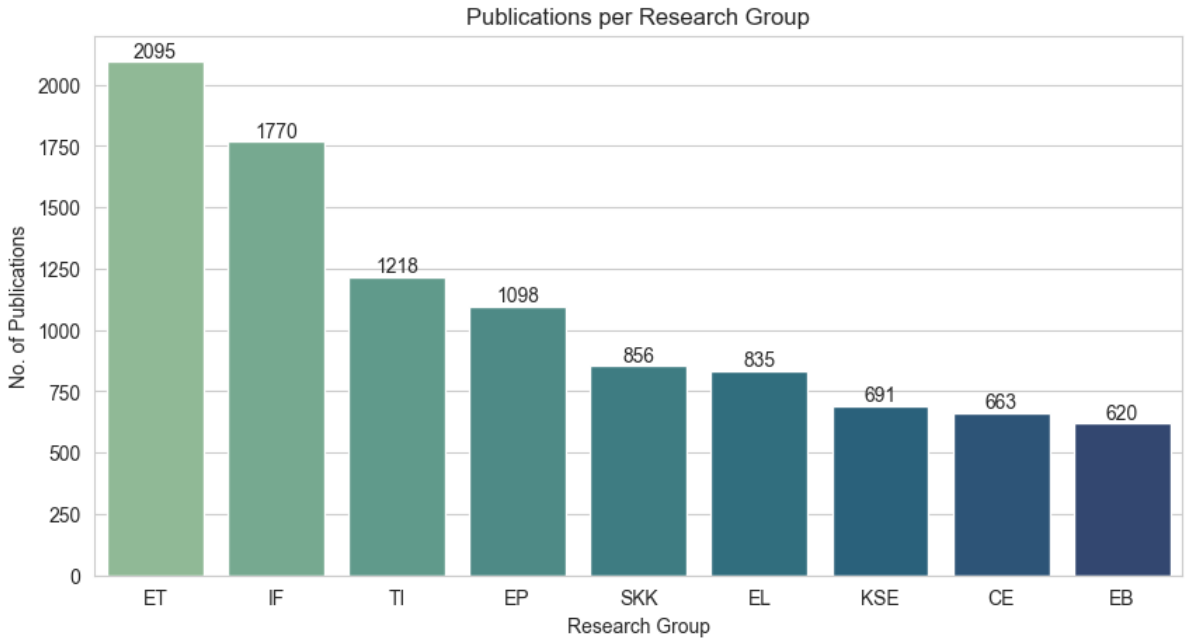

Fig 3. Number of publications per research group

2.2. Experiments

In this section, we explain our experiment scenario. The general steps for our document clustering flow could be seen on **Fig 4**.

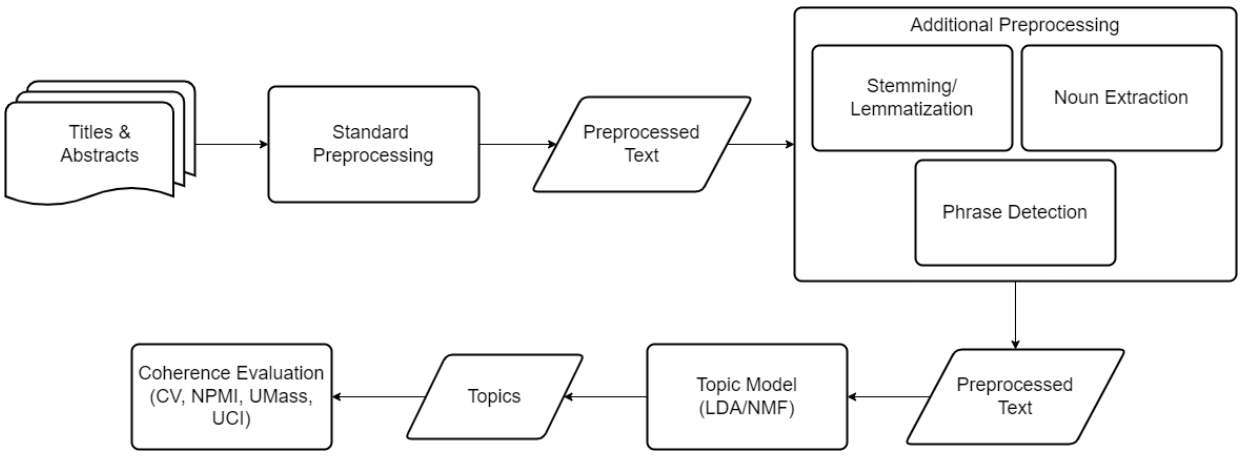

Fig 4. Initial Process of Experiment

In terms of scientific papers, we use just titles and abstracts (Kim & Gil, 2019) or full-text document (Terko et al., 2019) to perform clustering. Research (Syed & Spruit, 2017) stated that a collection of scientific articles from a specific domain would perform better with full-text data, while articles from a broader range would perform better with only the abstracts. Since our

data contains articles from mainly two knowledge areas (informatics and electrical engineering), we decided to use just the titles and abstracts.

Then, we performed basic preprocessing to the texts: lowercasing, removing punctuations and numbers, basic stop words removal (with stop words as listed in the NLTK library), translating non-English documents, and removal of duplicate entries. There are also non-paper publications in the corpus, for example, committee documents. As we are unsure of the importance of those documents, we decided to make two versions of our corpus: one with the non-paper documents included (non-filtered) and one with the non-paper documents excluded (filtered). We tested our models on both versions.

Then, an experiment is conducted to obtain the most optimal preprocessing pipeline. The purpose of this experiment is to discover which preprocess methods could improve performance and which methods hurt the performance instead. We tested a total of three preprocessing methods, as summarized on **Table 3**. Each of these methods will be explained in more detail in later sections.

Table 3. Preprocessing Steps to Evaluate

Name	Process Objective
Stemming, lemmatization and technical stop word removal	Removing noises & making sure words with the same meaning have the same representation
Noun extraction	Capturing only important entities & simplifying data
Phrase detection	Grouping concepts, e.g. “electrical engineering” vs “electrical” “engineering”

For each method, we performed a grid search (with doing method vs without doing method) and varying the types of documents (filtered vs unfiltered) and the type of the model (LDA vs NMF), which are then tested with coherence scores. Best methods are then included in the pipeline to build the final model, which we will analyse its results further.

2.2.1. Standard Preprocessing

For each method, we performed a grid search (with doing method vs without doing method) and varying the types of documents (filtered vs unfiltered) and the type of the model (LDA vs NMF), which are then tested with coherence scores. Best methods are then included in the pipeline to build the final model, which we will analyse its results further.

2.2.2. Pipelining Experiment

For each experiment, we would compare the performance of LDA and NMF of two topic modelling methods against filtered and unfiltered corpus. We used LDA and NMF models as they are less computationally expensive compared to deep learning methods and BERTopic (Grootendorst, 2022). As our corpus dataset is relatively small and consists of short texts, using data-hungry BERT (Devlin et al., 2018) algorithms may be less effective for this case study. Using LDA and NMF models also allow us to further analyse the formed cluster and gain more information about our dataset, as opposed to deep learning methods that use black box approach (Zini & Awad, 2023).

The parameters set for this experiment can be seen on **Table 3**. We set our initial number of topics as 9 to match the number of research groups in the faculty. Each experiment is evaluated by their coherence scores, namely CV, UMass, NPMI, and UCI. For top final models, we also examined their topic frequency and the top 5 words for each topic.

Table 3. Set Parameters for Experiments

Parameter	Value	Parameter	Value

n_topics:	9	iterations / max_iter:	10 000
random_state:	42	passes:	5

Experiment 1: Additional Preprocessing. This experiment aims to evaluate which preprocessing pipeline has the best effect for our cluster results. We tested three kinds of preprocessing methods: lemmatizing, stemming, and technical stop word removal.

Lemmatizing is turning the word into its root form, for example, *writing* and *written* becomes *write*. Stemming is cutting the word into a short form, for example *writing* and *written* becomes *writ*. Both stemming and lemmatization aim to make sure words with the same meaning do not have different representations, as in the previous example, writing and written should not be represented differently. Stemming is usually faster than lemmatization because lemmatization requires a dictionary look up, but stemming might cause confusion and ambiguity due to different words getting cut into the same representation. For example, stemming *universal* and *university* would produce the same representation (*univers*) despite those words having different meanings. Lemmatization would correctly differentiate universal and university but would need longer time due to the dictionary look up process. We conducted an experiment to decide whether it is better to use lemmatization or stemming for our corpus.

Technical stop words, in contrast with basic stop words, are the words that frequently occur in a domain. Stop words are the words which are removed before any text processing is executed because those words are insignificant and do not add any meaning (Rajaraman & Ullman, 2011). In general, stop words are filler words like *am*, *is*, *are*, *he*, *she*, etc. In this case study, our corpus mainly contains scientific papers, so there are new stop words relating to academia. For example, the words *methods*, *data*, and *paper* are not considered as stop words generally, but because our corpus is more specific towards academic publications, those words will appear in almost every document as they relate to experiment methods. Thus, those words became stop words as they do not add significant information about the text they appeared on. Those new stop words will be referred to as tech stop words.

Technical stop words detection has been researched in (Sarica & Luo, 2021). In the research, they analysed texts in the patent database to identify stop words in the engineering domain based on statistical measures. The final stop word list was constructed by creating sorted lists based on the statistical measures which then were evaluated by humans. The final list consists of 26 new stop words, combined with 62 words from previous study, they produced a list of 87 technical stop words.

After constructing the list, (Sarica & Luo, 2021) conducted a case study on multi-class classification with LSTM. The result showed that removing technical stop words increased the precision, recall, and accuracy scores compared to models trained with raw texts and trained with texts that had only basic stop words removed.

In this experiment, we performed a grid search on lemmatization vs. stemming the words and with vs. without technical stop words. Lemmatizing and stemming were carried out with the NLTK library. For the technical stop words, we used the strategy used in (Sarica & Luo, 2021) to identify the stop words in our corpus. We did not use the final list because our corpus included texts from the electrical engineering department which may have a different set of stop words than the one used in (Sarica & Luo, 2021). There are 4 combinations from the grid, multiplied by number of models (2) and corpus (2), we tested a total of 16 models in this experiment.

Experiment 2: Noun Extraction. This experiment aims to determine whether it is better to use full text or just noun phrases to cluster our documents. This is based on (Kim & Gil, 2019) where they removed stop words and extracted only nouns to reduce the number of processed texts and improve processing efficiency. We applied the same method to our experiment as most important words and concepts in our domain were nouns, e.g., internet of things, machine learning, robotics, signal processing, etc. For the implementation, we utilized SpaCy and liamca's noun phrase extraction algorithm (github.com/liamca/noun-phrase-extraction).

We also varied the phrasing method, where we will be checking whether it is better to group the phrased nouns (*New York* becomes "*new_york*") or separate them (*New York* becomes "*new*" and "*york*"). In this experiment, we would like to discover the significance of bigram or trigram phrase detection to our topic models, since these detections might provide detailed topic words. For example, the term *neural network* is widely used in deep learning-based research, however either *neural* or *network* term independently would give a different context to a document when being considered as a singular token.

Experiment 3: Phrase Detection. This experiment aims to determine whether it is better to process phrases as phrases or separate words. For example, it might be better to consider biomedical engineering as one concept (biomedical_engineering) instead of separate words (*biomedical* and *engineering*). Those sequences of words are often called *n*-grams, with *n* is the number of words in a sequence. The phrase *biomedical_engineering* is considered a bigram (2-gram).

N-gram detection for clustering texts were researched in (Mohemad et al., 2021). In the study, they clustered crime event related texts to group them into the five classes of its modus operandi (MO). They used phrase detection as a preprocessing step, and experiment-ed with three variations: 2-gram, 3-gram, and 4-gram. Results showed that detecting 2-gram and 3-gram did not improve the results, but 4-gram has the best performance, even exceeding baseline.

In this study, we experimented on three kinds of n-grams: 1-gram, 2-gram, and 3-grams. Even though (Mohemad et al., 2021) stated that 4-gram has best performance, we decided to just experiment on 1, 2, and 3-grams because 2 and 3-grams were more commonly used, while usage of 4-grams are quite rare. Also, (Mohemad et al., 2021) used data from a completely different domain, so it is safer to experiment on commonly used n-grams rather than the rare one.

Experiment 4: Final Model. After obtaining the best variations from each experiment, we conducted one more experiment to build a stronger model based on previous results. We will note which parameters from the experiment which give better results, then use those parameters on our fourth experiment.

After building the model, we conducted an elbow method to test whether there are a more fitting number of topics other than 9, because it could be that a paper in one research group could be divided further into more specific domains. We also looked into the keywords for each topic in the optimum model for further analysis.

3. Results and Discussion

3.1. Preliminary Experiment

The best results for experiment 1, 2, and 3 are summarized in **Table 4**, while full results can be seen on Appendix A.

Table 4. Result summary for experiment 1-3

#	params	CV	NPMI	UMass	UCI
1	NMF + filtered + lemma + tech stop words removed	0.554	0.062	-2.618	0.113
2.a	NMF + unfiltered + phrased	0.689	-0.134	-2.493	-4.589
2.b	NMF + unfiltered + unphrased	0.679	0.113	-2.421	0.705
3	LDA + unfiltered + 1-gram	0.494	0.047	-1.851	0.316

Experiments 1 and 2 showed that NMF is best for clustering our corpus, while experiment 3 performed best with LDA, but with a far lower score. Reference (Egger & Yu, 2022) stated that NMF is better compared to LDA when dealing with shorter texts. As our texts consist of only title and abstracts, not whole publication text, NMF is suitable because of the short length of the corpus.

Experiment 2 and 3 showed that it is better to use unfiltered texts (includes non-scientific documents), while experiment 1 showed the filtered corpus is better. We further investigated this by building two models, one with filtered texts and the other with unfiltered text and compared the results. Then, we analysed the result for each experiment step as follows.

Additional Preprocessing. From experiment 1, lemmatization performed better than stemming. This is because stemming is prone to cause ambiguity. As explained before, stemming reduced *universal* and *university* to the same representation *univers* despite the two words having different meanings; while lemmatization would represent the two words correctly because of the dictionary look up process. Our corpus has many similar words with different meanings, where stemming those words causes more ambiguity thus making it difficult for the model to cluster the documents correctly.

For technical stop words removal, we applied the algorithm from (Sarica & Luo, 2021) to our own corpus. This approach produced 88 technical stop words whose complete list is displayed on Appendix B. Study (Sarica & Luo, 2021) produced a similar number of new stop words (87), however their list consisted of more general words (mentioned, accordingly, furthermore, instead) while our list consisted of words relating to academia (process, performance, result, improve, technique, data). Despite having more specific words, stop word removal successfully improved our model scores, which means the identified words were truly words that add no additional meaning and removing them reduces the ambiguity those words cause.

From this experiment, we decided that lemmatization and tech stop words removal would be included in our final model pipeline.

Phrase Detection. Based on experiment 3, the model with 1-gram is performing better than model with 2 and 3-gram. We investigated the most frequent terms in our corpus after running the phrase detection function and found that the top-20 terms were mostly still in the form of 1-gram, with frequency at least 1500. The top-1 term, *system*, appeared around 8000 times. Compared to that, the most frequent 2-gram, *real_time*, is far behind by appearing only 571 times, while the most frequent 3-gram, *inspect_non_controlled*, is even more far behind with only 160 appearances.

This proves that despite detecting the phrases, they still appeared less than the commonly used 1-gram terms. As both LDA and NMF utilized word count (bag-of-words) in their algorithm, terms that appeared less would have less weight to determine what topic a document belongs to. This means that even though 2 and 3-grams were detected, they do not have much impact to improve model performance because they do not appear as frequently as 1-grams, hence having less weight compared to the 1-grams.

Instead, detecting 2 and 3-gram increased the number of new stop words. Among common 2 and 3-grams, we found phrases that do not add new information about the document topic, for example the phrases *paper_presents, case_study, design_implementation, proposed_method, result_show*. Even worse, these stop phrases appeared more often than the meaningful ones, so phrase detection adds more noise to the corpus. This could be improved by first removing the technical stop words then running phrase detection algorithm, but we concluded that it would not help as much because of the low n-gram appearance problem which was explained before. Thus, we decided to exclude phrase detection from our final model pipeline.

Noun Extraction. The model built with corpus that consisted of nouns only produced the highest CV scores, which means noun extraction is the most helpful among all preprocessing we experimented on. This proved that the most important words and concepts in our corpus are indeed nouns, so eliminating non-noun words removes noise and allows the model to focus only on important features. Noun extraction worked great, so we certainly included noun extraction in our final model pipeline.

However, there is little CV score difference between phrased and unphrased nouns. Just like the case in phrase detection, this could be happening because common phrases do not appear as often as common words (2 and 3-gram appeared less than 1-grams), so grouping phrases might have no significant difference on how the documents were clustered. Despite having similar CV scores, the difference for UCI scores were drastic: phrased corpus UCI score was significantly lower than the score for unphrased one. We decided to further investigate this issue by experimenting on both phrased and unphrased corpus when building our final model and later compared their topic results.

3.2. Final Model

Based on experiment 1-3, it is known that NMF, 1-gram, lemmatization, and tech stop words removal produced best results. However, phrased vs. unphrased words were tied and experiment 1 showed that it is best to use filtered texts, while experiment 2-3 showed that unfiltered texts is better.

Due to the score ties and inconsistency, we conducted an additional experiment for our final model. We ran a grid search with filtered text vs. unfiltered text and phrased vs. unphrased texts; there are four tested final models in total. We also conducted elbow methods to determine the optimum number of clusters for all four models. The results are summarized in **Table 5**.

Table 5. Result summary for experiment 4

#	phrased	filtered	n_cluster	UMass	CV	NPMI	UCI
1.a	yes	no	9	-2.493	0.689	-0.134	-4.589
1.b	yes	no	11	-2.493	0.681	-0.136	-4.491
2	yes	yes	18	-2.605	0.652	-0.120	-4.337
3	no	no	9	-2.421	0.679	0.113	0.705
4	no	yes	13	-2.724	0.653	0.103	0.510

From the table above, we noticed that filtered corpus needs a higher number of topics. This probably happened because filtered corpus excluded general documents such as conference committees and preface texts, leaving only technical documents and publications. Those documents are more specific to a domain, thus needing more topics to properly cluster them. We will further analyse the topics by comparison explained below.

In the meantime, we concluded that the best clustering model is constructed by NMF with lemmatized, stop word removed, phrased, and unfiltered corpus. It is best constructed with nine topics. For more details about the results, we further analysed model 1.a as it produced the best

numerical results. We also analysed close seconds, 1.b and 3, for comparisons. Firstly, topic words for model 1.a are shown on **Table 6**.

Table 6. Topic from model 1.a

ID	Topic words	Interpreted Topic	Number of documents
1a.1	output, input, current, motor, speed, controller, vehicle, experimental, low, component	robotics	1421
1a.2	management, framework, business, organization, case, government, activity, architecture, concept, important	IT enterprise / governance	819
1a.3	energy, case, load, renewable, source, cost, electricity, generation, potential, plant	renewable energy / green IT	1008
1a.4	indonesian, classification, best, word, machine, text, language, extraction, sentence, vector	NLP	811
1a.5	learning, student, architecture, education, processing, activity, teacher, machine, medium, game	e-learning	874
1a.6	rate, error, channel, parameter, bit, scheme, term, noise, low, wireless	wireless communication	1148
1a.7	voltage, characteristic, parameter, effect, experimental, current, material, discharge, partial, property	electrical / material experiments	1357
1a.8	device, internet, function, thing, sensor, human, mobile, smart, protocol, main	internet of things / smart device	1449
1a.9	substrate, dielectric, antenna, epoxy, characterization, fr4, dimension, thickness, structure, microstrip	antenna	731

The nine research groups in our department mainly can be divided into two bigger groups: electrical engineering (EL) and informatics (IF). The topics produced by model 1.a showed more topics from EL (topic 1, 6, 7, 8, 9) than IF (topic 2, 4, 5). This is natural as there are more research groups in EL than in IF. There is also a unique topic that combines renewable energy from EL and green IT from IF (topic 3), probably because both of the topics discuss the environment.

There are also several topics with large distributions (topic 1, 7, and 8) exceeding 1300. This probably happens because those topics are more general than the other topics like NLP (topic 4), e-learning (topic 5), or antenna (topic 9) which are more specific. Those big topics probably consisted of more diverse documents and likely the publications that do not fit into the more specific topic were classified into those big topics.

We chose n_topics = 9 because there are nine research groups, however the clustering result does not really match with our existing research groups. Several topics from a research group appeared more than once. For example, topics 1 and 7 are both topics from the electronics research group, while topics 6 and 9 are from telecommunication engineering. We do not have a cluster representing topics from biomedical engineering or electrical power engineering. This could happen due to the imbalance of number of publications in each research group, for example biomedical engineering is relatively new so there might not be as many publications from the research group. Another reason for this happening is because some research groups cover more topics than others, for example electronics could also cover some basic of electrical power engineering and control system & computer.

Next, we compared those topics with the topics produced by model 1.b. The list of topics is shown on **Table 7**. The model produced two more topics than model 1.a, but the numerical scores for both models are quite the same.

Table 7. Topics from model 1.b

ID	Topic Words	Interpreted Topic	Number of documents
1b.1	sensor, environment, monitoring, important, thing, internet, iot, mobile, dynamic, smart	IoT / smart device	467
1b.2	current, energy, voltage, load, output, electric, experimental, source, inverter, renewable	electrical power	1133
1b.3	learning, machine, classification, indonesian, word, best, language, text, extraction, neural	NLP	1080
1b.4	characteristic, voltage, parameter, discharge, partial, effect, material, insulation, pattern, important	electrical / material	949
1b.5	low, range, light, circuit, rate, standard, noise, receiver, modulation, source	signal processing / fiber optics	839
1b.6	function, controller, linear, speed, solution, position, cost, motor, error, platform	robotics	631
1b.7	processing, object, device, part, computer, architecture, digital, field, human, camera	computer vision	696
1b.8	learning, student, activity, education, internet, concept, medium, digital, experience, interaction	e-learning	968
1b.9	substrate, dielectric, antenna, epoxy, characterization, fr4, dimension, structure, thickness, microstrip	antenna	589
1b.10	case, management, framework, business, organization, government, solution, architecture, existing, tool	IT enterprise / governance	1030
1b.11	parameter, error, rate, channel, wireless, scheme, evaluation, access, term, transmission	wireless communication	1236

Compared to topics from model 1.a, we found several new topics; namely electrical power (1b.2), signal processing / fiber optics (1b.5), and computer vision (1b.7). The topic that appeared in 1.a but not in 1.b is renewable energy / green IT (1a.3).

The new topics might be formed due to the higher number of topics, so the model could classify the documents into more specific clusters. Hence, there are no big clusters with distributions over 1300. However, there is one very small cluster that consists of only 467 instances (1b.1, IoT), which is interesting because its counterpart from 1.a (1a.8, IoT) is the biggest of all clusters. This confirms that cluster 1a.8 consists of various documents slightly unrelated to IoT, which then can be broken down and formed several new topics.

Likewise, topic 1a.3 might disappear because the instances grouped in 1a.3 found a better suiting cluster in model 1.b. Thus, the cluster related to green IT grows bigger (1b.10, IT enterprise/governance with number of documents 1030) than its counterpart (1a.2, IT enterprise/governance with number of documents 819). On the other side, instances related to renewable energy were grouped into cluster 1b.2 (electrical power) which is a new topic. This clustering also made more sense because finding renewable energy is a common research topic for generating power.

Thus, despite having similar metric scores, model 1.b seemed to produce better clustering results by human judgement. However, model 1.b still does not include some minor research groups, like biomedical engineering or computer engineering, which is reasonable because the publication from those research groups is far smaller by number compared to the other research groups.

We also compared these results with our second runner-up model, which is model 3. This model also produced 9 topics but was conducted without phrasing the nouns. The scores for model 3 are also similar to model 1.a and 1.b except for the exceptionally low UCI. **Table 8** shows the topics from model 3.

Table 8. Topics from model 3

ID	Topic Words	Interpreted Topic	Number of documents
3.1	dielectric, substrate, structure, filter, epoxy, waveguide, fr4, response, microstrip, bandwidth	microwave / high-frequency systems	664
3.2	voltage, discharge, partial, oil, insulation, transformer, characteristic, electric, electrical, current	electrical systems	492
3.3	energy, current, load, renewable, electric, source, solar, plant, motor, hybrid	renewable energy	700
3.4	indonesian, classification, language, text, word, extraction, sentence, recognition, speech, based	NLP	1079
3.5	smart, device, sensor, mobile, monitoring, home, environment, protocol, platform, robot	smart devices / robotics	1206
3.6	channel, rate, low, error, estimation, voltage, scheme, output, input, controller	communication systems	1912
3.7	learning, student, game, machine, education, environment, mobile, activity, deep, language	e-learning / gamification / mobile	902
3.8	management, framework, architecture, business, government, digital, case, organization, enterprise, engineering	IT enterprise / governance	2182
3.9	antenna, array, patch, substrate, microstrip, radiation, radar, gain, bandwidth, epoxy	antenna / radar	481

Despite having the same number of topics as model 1.a, model 3 produced almost entirely different clusters. The only recurring topics with those produced in 1.a are electrical systems (3.2), NLP (3.4), and IT enterprise/governance (3.8). Interestingly, there is a wide gap in the distribution of those topics. For example, topic 3.8 has almost 2200 instances, while topic 3.2 consists of only 492 instances. Meanwhile, the new appearing topics seem to be a more specific version of those mentioned in 1.a and 1.b, while some other ones are like a mash up of several topics in 1.a and 1.b.

For example, topic 3.1 shared several keywords with topic 3.9. However, topic 3.1 was more focused on antennas and their design elements (proved with keywords *radar, substrate, epoxy, patch*), while topic 3.9 was focused more on high-frequency systems (proved with keywords *filter, waveguide, microstrip*). It might be better to combine those two topics as the two have subtle differences and discuss antenna / signal. Moreover, the two topics also have smaller instance count compared to other clusters, which proved that the two were indeed very specific.

As opposed to topic 3.1 and 3.9 which became very specific, we also found several topics that mashed up several domains in a cluster. Topic 3.5 seems to be clustering smart devices and robotics together, while in model 1.a and 1.b they were separated. This cluster still made sense because both smart devices and robotics have several aspects that collide, e.g., both need sensors and interact with the environment.

The more erroneous cluster happened on topic 3.7, where it seems to put e-learning and machine learning together. While both concepts include the word “learning”, they were two entirely different topics. Documents about e-learning were mostly about software engineering, constructing new applications, while machine learning is about finding patterns in data and

predicting patterns in unseen data. These erroneous clusters might be the cause why UCI score for model 3 increased significantly compared to UCI score for model 1.a and 1.b.

Based on our analysis above, we concluded that model 1.b, with phrased nouns, unfiltered documents, and eleven topics, was the best model for clustering lecturers' publications in STEI.

3.3. Result Validation

We sampled 200 random publications from our STEI dataset and manually labelled them to the topic group from our model that we think the publication belongs to. We then ran the clustering pipeline to the two versions of the sample: the raw data, and the data after performing noun detection, phrasing, and technical stop words removal. For fair comparison, we performed lowercasing, punctuation and general stop word removal, and lemmatization to both versions. We evaluated the results, which are shown in **Table 9**.

Performing noun detection, phrasing, and technical stop words increases the accuracy of our sample from 71.1% to 80.7%. In comparison, previous research (Sarica & Luo, 2021) applied stop words removal for a clustering task and reached an accuracy of 95.9% for dataset with general stop words removed and 97.0% for the dataset with general and technical stop words removed. The summary of this comparison is also shown on **Table 9**.

While the previous research achieved higher accuracy, our approach presented more improvements from the baseline, proving that noun detection, phrasing, and technical stop words removal can effectively improve clustering performance for scientific texts.

Table 9. Metric comparison between this and previous research

Evaluated texts	Precision	Recall	Accuracy
This research			
General stop words removed	0.710	0.715	0.711
Noun detection, phrasing, general and technical stop words removed	0.815	0.807	0.807
(Sarica & Luo, 2021)			
General stop word removed	0.961	0.959	0.959
General and technical stop words removed	0.971	0.970	0.970

We also performed our pipeline to a sampled arXiv dataset. For this experiment, we randomly sampled 100 papers from the category of computer science and 100 papers from the category of electrical engineering and systems science to mimic the nature of our dataset. We also limited the time frame to only include papers published in 2020 to 2024.

To compare with the performance on our dataset, we used Hellinger distance which measures the similarity between documents from the same topic and to other topics. This metric calculates the distance between two probability distributions, which makes it suitable for topic vectors of LDA or NMF topic models (Muchene & Safari, 2021).

We computed the Hellinger similarity between topics by calculating the distance between every document pairs of different topics and averaged their scores. The Hellinger similarity is the complement of the Hellinger distance, which value falls between 0 and 1. Two topics will be considered similar when its Hellinger similarity is closer to zero, and vice versa. The results can be seen in **Fig 6**.

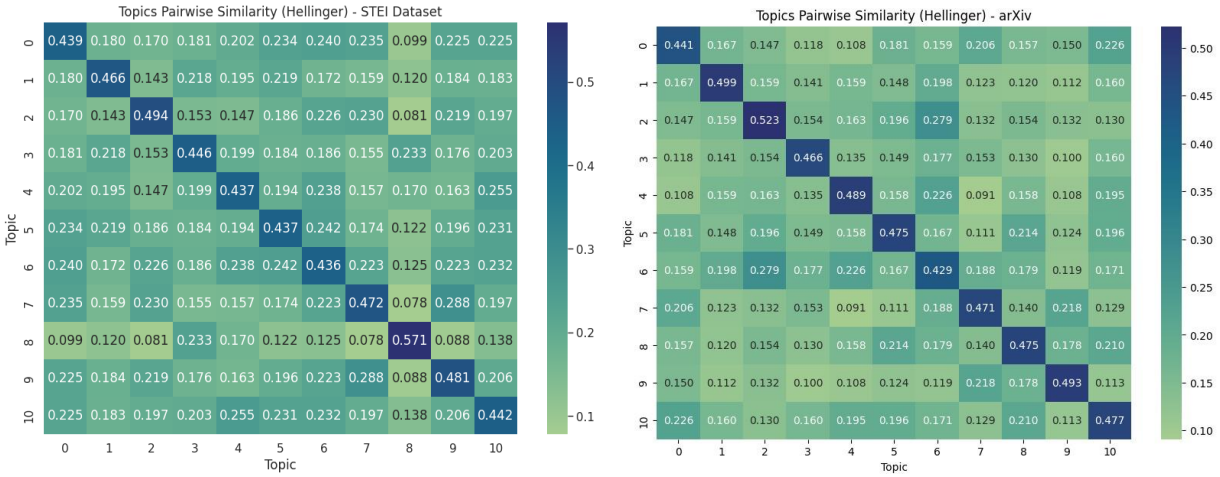

Fig 6. Hellinger similarity for STEI dataset (left) and arXiv dataset (right)

From this metric, we observed that documents within each groups have few overlapping topics between each other. This is shown by the relatively lower similarity score between document groups. We noticed that documents on the same group also relatively stronger similarity score, although relatively speaking, its value far from a strong similarity, which is closer to one. This observation is similar on both our STEI and the sampled arXiv text documents.

4. Conclusions

In this study, we explored combinations of several text processing methods while addressing the challenge of clustering texts from specific domains: increased technical stop words and the use of similar terms for different concepts. Our findings indicate that the Non-Negative Matrix Factorization (NMF) model combined with lemmatization, technical stop word removal, noun extraction, and phrase detection performed best among all the combinations we tested. This model effectively grouped our institution's scientific documents into eleven clusters and improved the sample clustering results accuracy from 71.1% to 80.7%. We have also applied it to another dataset and confirmed that our method is generalizable, as the results were similar to when applied to our dataset.

The results have practical implications for improving literature retrieval effectiveness, discovering emerging research trends within an institute, and automating literature labelling. However, our study focused on a corpus from a single institution, specifically in electrical engineering and informatics, which may limit its applicability to other domains or interdisciplinary studies. Future research could explore the application of this methodology to a broader range of other scientific fields other than engineering.

Acknowledgments

This work was supported by the P2MIGB Grant No. 968/IT1.C12/KU/2023 provided by the School of Electrical Engineering and Informatics, ITB.

Conflict of Interest

The authors declare no conflicts of interest.

References

Aftab, F., Bazai, S. U., Marjan, S., Baloch, L., Aslam, S., Amphawan, A., & Neo, T. K. (2023). A Comprehensive Survey on Sentiment Analysis Techniques. *International Journal of Technology*, *14*(6), 1288. https://doi.org/10.14716/ijtech.v14i6.6632

Bellaouar, S., Bellaouar, M. M., & Ghada, I. E. (2021). Topic Modeling: Comparison of LSA and LDA on Scientific Publications. *2021 4th International Conference on Data Storage and Data Engineering*, 59–64. https://doi.org/10.1145/3456146.3456156

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. *The Journal of Machine Learning Research*, *3*, 993–1022.

Chang, I.-C., Yu, T.-K., Chang, Y.-J., & Yu, T.-Y. (2021). Applying Text Mining, Clustering Analysis, and Latent Dirichlet Allocation Techniques for Topic Classification of Environmental Education Journals. *Sustainability*, *13*(19), 10856. https://doi.org/10.3390/su131910856

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. http://arxiv.org/abs/1810.04805

Egger, R., & Yu, J. (2022). A Topic Modeling Comparison Between LDA, NMF, Top2Vec, and BERTopic to Demystify Twitter Posts. *Frontiers in Sociology*, *7*. https://doi.org/10.3389/fsoc.2022.886498

Grootendorst, M. (2022). *BERTopic: Neural topic modeling with a class-based TF-IDF procedure*. http://arxiv.org/abs/2203.05794

Hadiat, A. R. (2022). *Topic Modeling Evaluations: The Relationship Between Coherency and Accuracy* [University of Groningen]. https://fse.studenttheses.ub.rug.nl/28618/1/s2863685_alfiuddin_hadiat_CCS_thesis.pdf

Hassani, A., Iranmanesh, A., & Mansouri, N. (2021). Text mining using nonnegative matrix factorization and latent semantic analysis. *Neural Computing and Applications*, *33*(20), 13745–13766. https://doi.org/10.1007/s00521-021-06014-6

Janmaijaya, M., Shukla, A. K., Muhuri, P. K., & Abraham, A. (2021). Industry 4.0: Latent Dirichlet Allocation and clustering based theme identification of bibliography. *Engineering Applications of Artificial Intelligence*, *103*, 104280. https://doi.org/10.1016/j.engappai.2021.104280

Jelodar, H., Wang, Y., Yuan, C., Feng, X., Jiang, X., Li, Y., & Zhao, L. (2019). Latent Dirichlet allocation (LDA) and topic modeling: models, applications, a survey. *Multimedia Tools and Applications*, *78*(11), 15169–15211. https://doi.org/10.1007/s11042-018-6894-4

Kadhim, A. I. (2019). Survey on supervised machine learning techniques for automatic text classification. *Artificial Intelligence Review*, *52*(1), 273–292. https://doi.org/10.1007/s10462-018-09677-1

Kim, S.-W., & Gil, J.-M. (2019). Research paper classification systems based on TF-IDF and LDA schemes. *Human-Centric Computing and Information Sciences*, *9*(1), 30. https://doi.org/10.1186/s13673-019-0192-7

Larsen, P. O., & von Ins, M. (2010). The rate of growth in scientific publication and the decline in coverage provided by Science Citation Index. *Scientometrics*, *84*(3), 575–603. https://doi.org/10.1007/s11192-010-0202-z

Laxmi Lydia, E., Krishna Kumar, P., Shankar, K., Lakshmanaprabu, S. K., Vidhyavathi, R. M., & Maseleno, A. (2020). Charismatic Document Clustering Through Novel K-Means Non-negative Matrix Factorization (KNMF) Algorithm Using Key Phrase Extraction. *International Journal of Parallel Programming*, *48*(3), 496–514. https://doi.org/10.1007/s10766-018-0591-9

Lee, D. D., & Seung, H. S. (1999). Learning the parts of objects by non-negative matrix factorization. *Nature*, *401*(6755), 788–791. https://doi.org/10.1038/44565

Leung, X. Y., Sun, J., & Bai, B. (2017). Bibliometrics of social media research: A co-citation and co-word analysis. *International Journal of Hospitality Management*, *66*, 35–45. https://doi.org/10.1016/j.ijhm.2017.06.012

Li, Y., Wang, K., Xiao, Y., & Froyd, J. E. (2020). Research and trends in STEM education: a systematic review of journal publications. *International Journal of STEM Education*, *7*(1), 11. https://doi.org/10.1186/s40594-020-00207-6

Lubis, F. F., Mutaqin, M., Putri, A., Waskita, D., Sulistyaningtyas, T., Arman, A. A., & Rosmansyah, Y. (2021). Automated Short-Answer Grading using Semantic Similarity based on Word Embedding. *International Journal of Technology*, *12*(3), 571. https://doi.org/10.14716/ijtech.v12i3.4651

M. Mohammed, S., Jacksi, K., & R. M. Zeebaree, S. (2021). A state-of-the-art survey on semantic similarity for document clustering using GloVe and density-based algorithms. *Indonesian Journal of Electrical Engineering and Computer Science*, *22*(1), 552. https://doi.org/10.11591/ijeecs.v22.i1.pp552-562

Mehta, V., Bawa, S., & Singh, J. (2021). WEClustering: word embeddings based text clustering technique for large datasets. *Complex & Intelligent Systems*, *7*(6), 3211–3224. https://doi.org/10.1007/s40747-021-00512-9

Mifrah, S. (2020). Topic Modeling Coherence: A Comparative Study between LDA and NMF Models using COVID'19 Corpus. *International Journal of Advanced Trends in Computer Science and Engineering*, *9*(4), 5756–5761. https://doi.org/10.30534/ijatcse/2020/231942020

Mohemad, R., Muhait, N. N. M., Noor, N. M. M., & Othman, Z. A. (2021). The Impact of N-gram on the Malay Text Document Clustering. *Malaysian Journal of Information and Communication Technology*, *6*(2), 22–29.

Muchene, L., & Safari, W. (2021). Two-stage topic modelling of scientific publications: A case study of University of Nairobi, Kenya. *PLOS ONE*, *16*(1), e0243208. https://doi.org/10.1371/journal.pone.0243208

Pavithra, & Savitha. (2024). Topic Modeling for Evolving Textual Data Using LDA, HDP, NMF, BERTOPIC, and DTM With a Focus on Research Papers. *Journal of Technology and Informatics (JoTI)*, *5*(2), 53–63. https://doi.org/10.37802/joti.v5i2.618

Preetham M C, S., Reddy, B. R., Tharun Reddy, D. S., & Gupta, D. (2022). Comparative Analysis of Research Papers Categorization using LDA and NMF Approaches. *2022 IEEE North Karnataka Subsection Flagship International Conference (NKCon)*, 1–7. https://doi.org/10.1109/NKCon56289.2022.10127059

Rajaraman, A., & Ullman, J. (2011). Data Mining. In *Mining of Massive Datasets* (pp. 1–17). Cambridge University Press. https://doi.org/10.1017/CBO9781139058452.002

Sajid, N. A., Ahmad, M., Afzal, M. T., & Atta-ur-Rahman. (2021). Exploiting Papers' Reference's Section for Multi-Label Computer Science Research Papers' Classification. *Journal of Information & Knowledge Management*, *20*(01), 2150004. https://doi.org/10.1142/S0219649221500040

Sarica, S., & Luo, J. (2021). Stopwords in technical language processing. *PLOS ONE*, *16*(8), e0254937. https://doi.org/10.1371/journal.pone.0254937

Shah, N., & Mahajan, S. (2012). Document Clustering: A Detailed Review. *International Journal of Applied Information Systems*, *4*(5), 30–38. https://d1wqtxts1xzle7.cloudfront.net/81705889/ijais12-450691-libre.pdf

Shahnaz, F., Berry, M. W., Pauca, V. P., & Plemmons, R. J. (2006). Document clustering using nonnegative matrix factorization. *Information Processing & Management*, *42*(2), 373–386. https://doi.org/10.1016/j.ipm.2004.11.005

Smail, B., Aliane, H., & Abdeldjalil, O. (2023). Using an explicit query and a topic model for scientific article recommendation. *Education and Information Technologies*, *28*(12), 15657–15670. https://doi.org/10.1007/s10639-023-11817-2

Surjandari, I., Dhini, A., Wibisana, N., & Lumbantobing, E. W. I. (2015). University Research Theme Mapping: A Co-word Analysis of Scientific Publications. *International Journal of Technology*, *6*(3), 410. https://doi.org/10.14716/ijtech.v6i3.1462

Syed, S., & Spruit, M. (2017). Full-Text or abstract? Examining topic coherence scores using latent dirichlet allocation. *Proceedings - 2017 International Conference on Data Science and Advanced Analytics, DSAA 2017*, *2018-January*, 165–174. https://doi.org/10.1109/DSAA.2017.61

Terko, A., Zunic, E., & Donko, D. (2019). NeurIPS Conference Papers Classification Based on Topic Modeling. *2019 XXVII International Conference on Information, Communication and Automation Technologies (ICAT)*, 1–5. https://doi.org/10.1109/ICAT47117.2019.8938961

Tey, W.-L., Goh, H.-N., Lim, A. H.-L., & Phang, C.-K. (2023). Pre- and Post-Depressive Detection using Deep Learning and Textual-based Features. *International Journal of Technology*, *14*(6), 1334. https://doi.org/10.14716/ijtech.v14i6.6648

Tsuge, S., Shishibori, M., Kuroiwa, S., & Kita, K. (2001). Dimensionality reduction using non-negative matrix factorization for information retrieval. *2001 IEEE International Conference on Systems, Man and Cybernetics. e-Systems and e-Man for Cybernetics in Cyberspace (Cat.No.01CH37236)*, *2*, 960–965. https://doi.org/10.1109/ICSMC.2001.973042

Vayansky, I., & Kumar, S. A. P. (2020). A review of topic modeling methods. *Information Systems*, *94*, 101582. https://doi.org/10.1016/j.is.2020.101582

Wang, Y.-X., & Zhang, Y.-J. (2013). Nonnegative Matrix Factorization: A Comprehensive Review. *IEEE Transactions on Knowledge and Data Engineering*, *25*(6), 1336–1353. https://doi.org/10.1109/TKDE.2012.51

Yu, D., & Xiang, B. (2023). Discovering topics and trends in the field of Artificial Intelligence: Using LDA topic modeling. *Expert Systems with Applications*, *225*, 120114. https://doi.org/10.1016/j.eswa.2023.120114

Zibani, P., Rajkoomar, M., & Naicker, N. (2022). A systematic review of faculty research repositories at higher education institutions. *Digital Library Perspectives*, *38*(2), 237–248. https://doi.org/10.1108/DLP-04-2021-0035

Zini, J. El, & Awad, M. (2023). On the Explainability of Natural Language Processing Deep Models. *ACM Computing Surveys*, *55*(5), 1–31. https://doi.org/10.1145/3529755

Appendix A: Full Experiment Results

Step 1

id	topic model	process	filtered ?	tech stop words	num topics	cv	npmi	umass	uci
0	lda	lemma	TRUE	FALSE	9	0.378	0.007	-2.033	-0.278
1	lda	lemma	TRUE	TRUE	9	0.451	0.008	-3.546	-0.782
2	lda	stem	TRUE	FALSE	9	0.381	0.011	-2.126	-0.293
3	lda	stem	TRUE	TRUE	9	0.462	0.025	-2.514	-0.186
4	nmf	lemma	TRUE	FALSE	9	0.378	0.014	-1.740	0.052
5	nmf	lemma	TRUE	TRUE	9	0.554	0.062	-2.618	0.113
6	nmf	stem	TRUE	FALSE	9	0.420	0.024	-1.770	0.033
7	nmf	stem	TRUE	TRUE	9	0.529	0.062	-2.066	0.394
8	lda	lemma	FALSE	FALSE	9	0.440	0.016	-2.385	-0.310
9	lda	lemma	FALSE	TRUE	9	0.461	0.001	-3.639	-0.989
10	lda	stem	FALSE	FALSE	9	0.450	0.026	-2.003	-0.033
11	lda	stem	FALSE	TRUE	9	0.446	-0.001	-3.325	-0.843
12	nmf	lemma	FALSE	FALSE	9	0.389	0.001	-2.066	-0.458
13	nmf	lemma	FALSE	TRUE	9	0.534	0.037	-2.838	-0.588
14	nmf	stem	FALSE	FALSE	9	0.417	0.015	-1.891	-0.211
15	nmf	stem	FALSE	TRUE	9	0.510	0.043	-2.388	-0.134

Step 2

id	topic model	phrasing	filtered dataset	num topics	cv	npmi	umass	uci
0	lda	FALSE	TRUE	9	0.479	0.037	-3.284	-0.255
1	lda	TRUE	TRUE	9	0.527	-0.174	-3.580	-5.431
2	lda	FALSE	FALSE	9	0.481	0.025	-3.726	-0.636
3	lda	TRUE	FALSE	9	0.587	-0.185	-5.947	-5.865
4	nmf	FALSE	TRUE	9	0.578	0.072	-3.032	0.046
5	nmf	TRUE	TRUE	9	0.565	-0.143	-2.657	-4.570
6	nmf	FALSE	FALSE	9	0.679	0.113	-2.421	0.705
7	nmf	TRUE	FALSE	9	0.689	-0.134	-2.493	-4.589

Step 3

id	topic model	n-gram	filtered dataset	num topics	cv	npmi	umass	uci
1	lda	1	TRUE	9	0.464	0.038	-1.826	0.219
2	lda	2	TRUE	9	0.416	0.023	-1.914	0.053
3	lda	3	TRUE	9	0.402	0.009	-2.377	-0.249
4	lda	1	FALSE	9	0.494	0.047	-1.851	0.316
5	lda	2	FALSE	9	0.459	0.037	-1.936	0.234
6	lda	3	FALSE	9	0.427	0.029	-1.855	0.170
7	nmf	1	TRUE	9	0.464	0.037	-1.892	0.173
8	nmf	2	TRUE	9	0.408	0.021	-1.960	0.074
9	nmf	3	TRUE	9	0.414	0.023	-1.951	0.084
10	nmf	1	FALSE	9	0.479	0.040	-1.940	0.171
11	nmf	2	FALSE	9	0.422	0.023	-2.035	0.003
12	nmf	3	FALSE	9	0.418	0.026	-1.962	0.093

Appendix B: List of Identified Technical Stop Words

process	accuracy	analysis	present
level	compare	one	technique
develop	detection	feature	data
measure	perform	order	two
network	new	learn	study
increase	apply	however	development
conduct	require	service	determine
many	provide	performance	high
power	work	result	also
various	different	improve	type
need	make	signal	frequency
number	simulation	approach	method
propose	problem	research	several
indonesia	methods	user	design
applications	experiment	implement	test
use	system	show	obtain
algorithm	technology	time	information
reduce	-	find	model
build	quality	communication	value
software	implementation	measurement	support
base	condition	image	control
systems	paper	area	application