The Value of Pre-training for Scientific Text Similarity: Evidence from Matching Grant Proposals to Reviewers

Gabriel Okasa and **Anne Jorstad**
Data Team, Swiss National Science Foundation, Berne, Switzerland
gabriel.okasa@snf.ch and anne.jorstad@snf.ch

Abstract

Matching grant proposals to reviewers is a core task for research funding agencies. We approach this task as a text similarity problem to allow pre-filtering of a relevant subset of potential matches using pre-trained language models. Given the scientific nature of our English text corpus, we investigate the value of targeted pre-training of BERT models towards scientific documents for the matching task based on the text similarity. We benchmark the performance of BERT models with a classical bag-of-words approach using TF-IDF. The results reveal a clear benefit from pre-training BERT on scientific texts and additionally augmenting by citation graphs. Interestingly, the BERT models do not substantially out-perform TF-IDF on the texts from any discipline. The results are robust to various types of input data and modelling choices.

1 Introduction

The role of research funding agencies is to support scientific research by evaluating grant proposals and deciding which of them are eligible for funding. As a part of the evaluation procedure, submitted grant proposals need to be assigned to suitable reviewers who assess the scientific quality of the proposals (Hettich and Pazzani, 2006). Matching proposals to reviewers is, however, a very time-consuming task which requires scientific officers to manually screen available reviewers and assess their suitability to review given proposals. Such a matching process involves reading grant proposals, reading published works from reviewers, and in a consistent manner determining their similarity.

In order to support this matching procedure, we approach this task as a text similarity problem to leverage the benefits of natural language processing to pre-filter a subset of suitable reviewers. In particular, we use NLP models to vectorize the English texts of proposals and those of reviewers' publications. We then compute a text similarity measure between the proposals and reviewers' publications. For each proposal we rank-order the similarity scores of all potential reviewers to retrieve the subset of best-matching reviewers. This subset then serves the scientific officers as a pre-filtered pool of suitable reviewers. Such pre-filtering substantially reduces the time needed to screen all possible reviewers and helps to more efficiently allocate the resources of the scientific officers. Similar NLP-based approaches of matching proposals to reviewers have been suggested in the domains of grant and journal peer review (Hettich and Pazzani, 2006; Stelmakh et al., 2021) as well as scientific conferences (Charlin and Zemel, 2013) and also from the big bibliometric databases (e.g. Dimensions, SpringerNature, Elsevier).

For the vectorization of the texts of proposals and reviewers' publications, we contrast a bag-of-words approach using the TF-IDF (Term Frequency - Inverse Document Frequency) weighting (Spärck Jones, 1972) with a word embeddings approach using pre-trained transformer models (Vaswani et al., 2017). In comparison to TF-IDF, transformers produce contextualized text embeddings thanks to their self-attention mechanism. Transformer models became widely used for semantic text similarity tasks (Reimers and Gurevych, 2019; Yang et al., 2020; Chandrasekaran and Mago, 2021), even though simple bag-of-words methods such as TF-IDF often perform equally well (Shahmirzadi et al., 2019). Given the vast amount of open-source pre-trained language models available (Wolf et al., 2020), the choice of a suitable model for a given setting is *a priori* not clear. Due to the specific scientific domain of the grant proposal texts as well as reviewers' publication texts, we focus on models pre-trained specifically on scientific texts in English and investigate the value added by such targeted pre-training in comparison to a model pre-trained on a general text corpus. As such, we consider the BERT model (Devlin et al.,

2018), being one of the most popular open-source pre-trained models, as our baseline model. We compare BERT to SciBERT (Beltagy et al., 2019), the BERT extension pre-trained additionally on scientific texts as well as to SPECTER (Cohan et al., 2020), which is a further extension of SciBERT via citation graph augmentation. In particular, we use SPECTER2 (Singh et al., 2022), an updated version of the original SPECTER model. As a benchmark model we consider the TF-IDF weighting. For each of the considered models, we vectorize the texts for both grant proposals and reviewers' publications and compute their cosine similarities. Based on the rank-ordered similarities, we select a subset of best matching reviewers for each grant proposal. As such, we effectively build a *recommender system* based on text similarities. We evaluate the performance of the models by contrasting the subset of best-matching reviewers with the actual reviewer matching based on a manual assignment by scientific officers.

The results reveal a clear pattern in favor of models with targeted pre-training on scientific texts. We observe substantially better performance of SciBERT in comparison to BERT, while SPECTER2 also considerably outperforms SciBERT. These findings provide clear evidence for the value added by targeted pre-training of base models on a specific text corpus for a matching/recommendation task based on text similarity. In particular, additional pre-training of BERT on scientific texts improves the overlap between the manually matched and model-generated subset of reviewers. In addition to pre-training on scientific texts, incorporating the inter-document relatedness via citation graph further improves the overlap. Despite the clear improvements of scientific pre-training of the BERT model, only the most sophisticated one, i.e. the SPECTER2, clearly outperforms the TF-IDF model. These results are robust to changes in the types of text data inputs such as title and abstract as well as the amount of text data provided. Furthermore, the results do not depend on specific modelling choices and are robust to changes in the text embedding extraction such as mean pooling or CLS tokens for BERT models and uni-grams or n-grams for the TF-IDF model.

The code for the conducted analyses is publicly available at https://github.com/snsf-data/snsf-grant-similarity.[1]

[1]Due to data protection laws, the data cannot be shared.

2 Institutional Setting

Based on a government mandate, the Swiss National Science Foundation (SNSF) supports scientific research in all academic disciplines. The SNSF is the leading Swiss organisation for the promotion of scientific research. The main role of the SNSF is the evaluation of scientific grant proposals; those that are evaluated to be the best are awarded research funding. Within the evaluation procedure, the SNSF relies on external peer-reviewers as well as on internal reviewers in the form of members of the evaluation panels. In this study, we focus on the latter evaluation step. For each evaluation panel, the grant proposals need to be matched to at least 2 reviewers from a pre-defined pool of available reviewers. These panel reviewers then assess the quality of the grant proposal based on the external peer reviews and their own evaluation of the proposals. In order to warrant fair and professional evaluation, the reviewers should have sufficient expertise in the fields of research of the respective grant proposals.

The matching of grant proposals to reviewers requires scientific officers to manually screen the grant proposal texts and the texts of reviewers' publications. Such a procedure is feasible if the number of proposals and reviewers is limited. However, it poses a great challenge as the number of proposals and potential reviewers grows. In order to reduce the manual labor, we approach the matching procedure as a text similarity problem. We leverage the benefits of the NLP models to vectorize the English texts from grant proposals and texts from reviewers' publications and compute their text similarities via cosine distance. For each grant proposal, we rank-order the similarity scores and select a subset of best-matching reviewers. In other words, we build a recommender system based on text similarities. We further need to take additional constraints into account, such as conflicts of interest and a maximum workload per reviewer. Finally, the suggested matching of proposals to reviewers is validated and approved by scientific officers before the final assignment takes place. This procedure can be summarized in the following steps:

1. Download publication metadata for each reviewer from a bibliometric database
2. Vectorize texts of reviewers and proposals
3. Match reviewers to proposals based on the highest text similarity

4. Balance number of proposals across reviewers

5. Validate matching results by scientific officers

In this paper, we focus on the above steps 2 and 3 and investigate the value of pre-training transformer models targeted towards the scientific domain in contrast to a simple bag-of-words approach, in order to determine the most efficient method of pre-filtering suitable reviewers.

3 Data

In general, it is challenging to objectively evaluate the performance of text vectorization methods for text similarity tasks as we cannot directly observe the true underlying text similarity (Reimers et al., 2016; Shah, 2022). In order to overcome this challenge, we evaluate the recommendations based on the text similarity and rely on a manually annotated dataset of matched reviewers provided by the scientific officers from the SNSF. In particular, we use the data from the Postdoc.Mobility funding scheme from the August 2021 call. Postdoc.Mobility fellowships enable early career researchers who have recently completed their doctorates and would like to pursue a scientific or academic career in Switzerland to conduct research projects abroad for up to two years. The data includes 398 submitted grant proposals across disciplines, and a pool of 150 potential reviewers, making it an arguably representative case. For each grant proposal, we observe the first-best and second-best reviewer according to the best knowledge of the scientific officers.[2] Most importantly, this matching does truly reflect the best possible assignment as it does not consider any additional constraints such as conflicts of interest or workload limits to manipulate the final assignment. Thus it can be used as a validation for evaluating the recommendations for matching based on the underlying similarity between the grant proposals and reviewer's publications.

To assess the text similarity between the grant proposals and reviewer's publications, we rely on the text of titles and abstracts. Titles and abstracts are often used for semantic text similarity tasks, especially in the scientific domain (Cohan et al., 2020) and should provide a condensed summary of the most important aspects of a scientific text. For grant proposals we retrieve the titles and abstracts directly from the submitted proposal documents. For potential reviewers we download the titles and abstracts from their scientific publications from a bibliometric database.[3] To ensure a clean evaluation setup we restrict the texts of titles and abstracts to English texts only, for both proposals and publications, and keep only those reviewers with at least 10 English publications available in the database. This leaves us with a set of 320 grant proposals and 125 potential reviewers.[4] Table 1 below provides an overview of the data based on the research areas:[5]

Area	# Proposals (%)	# Reviewers (%)
SSH	50 (15.6)	20 (16.0)
MINT	147 (46.0)	62 (49.6)
LS	123 (38.4)	43 (34.4)

Table 1: Overview of Research Area Distribution

To investigate the influence of data inputs on the matching results, we vary the inputs along two dimensions. First, we vary the composition of the text data and compare the matching results based on 1) titles, 2) abstracts, and 3) concatenation of titles and abstracts, to explore the value of the particular types of texts. Second, we vary the amount of the text data and compare the publications from the last 5 years vs. publications from the last 10 years, to examine the importance of the publications' recency. Note that although on average the increase in number of publications is proportionate to the recency of the publications, there is a lot of heterogeneity as well. Additionally, due to differences in publication practices, the actual number of publications varies substantially across disciplines.[6]

4 Methods

Since their introduction, transformer models (Vaswani et al., 2017) have gained considerable

[2]The assignment by scientific officers has been done in accordance with the research area, whereas we do not restrict the model-generated assignment as such.

[3]The present analysis uses data from the Scopus database of Elsevier. In the future the SNSF will base its matchings on the Dimensions database.

[4]These restrictions concern predominantly proposals and reviewers from the disciplines of social sciences and humanities due to the diverse type of outputs in these disciplines that are covered less completely in bibliometric databases.

[5]We follow the official discipline classification of the SNSF and distinguish between three high-level research areas: Human and Social Sciences (SSH), Mathematics, Natural- and Engineering Sciences (MINT), and Biology and Medicine (LS).

[6]The average number of publications per referee is 44.7 for the last 5 years of record and 82.0 for the last 10 years. For differences in research areas, see Table 3 in Appendix.

attention in the field of applied natural language processing (Tunstall et al., 2022). One of the key innovations of the transformer architecture is the self-attention mechanism, which helps to capture the context within the input sequence (Turner, 2023). As such, transformers provide a text vectorization in a form of *contextualized* text embeddings. Such contextualized embeddings can be used for a variety of NLP tasks, including semantic text similarity (Chandrasekaran and Mago, 2021). Furthermore, the availability of open-source pre-trained models on platforms such as Hugging Face makes it convenient to deploy these models for a particular application (see e.g. Wolf et al., 2020).

In this study, we focus on the BERT-type models (Devlin et al., 2018), i.e. deep bidirectional transformers, which have gained large popularity for a variety of applied NLP tasks. The BERT models are pre-trained on large text corpus via bidirectional representations, conditioning on both left and right context in the text sequence in all layers of the model (Devlin et al., 2018). The text corpus for the pre-training of the base BERT model consists of the BookCorpus (Zhu et al., 2015) and the English Wikipedia. Given the specific scientific domain in our setting of grant proposals and reviewers' publications we compare the base BERT model with its extended version that used additional scientific texts from SemanticScholar for pre-training, the so-called SciBERT (Beltagy et al., 2019), as well as with a further extension of the SciBERT itself - the SPECTER2 (Singh et al., 2022) - which has been further augmented by citation graph in its pre-training to capture the inter-document relatedness.

As collecting and labelling pairs of grant proposal text data for specific fine-tuning of the models is costly and often infeasible in practice due to the limited resources of the scientific officers, we focus on evaluating the pre-trained models as given, *without* additional fine-tuning. By doing so, we can effectively assess the value added by the specific pre-training of these models targeted towards scientific texts and their suitability for a matching/recommendation task based on the scientific text similarity. The pre-trained models as such can be used off-the-shelf for extracting the text embeddings via the so-called CLS token from the last hidden layer of the network, a classification token that provides an aggregate representation of the text sequence (Devlin et al., 2018; Cohan et al., 2020). An alternative representation for the text sequence can be obtained by the so-called mean pooling, which averages all 512 tokens from the last hidden layer to get the text embedding. Such extractions of the embeddings from pre-trained models is common for a variety of NLP tasks (Kjell et al., 2023; Wu et al., 2023) as well as for text similarity in particular (May et al., 2019; Zhang et al., 2019; Qiao et al., 2019), although it has been pointed out by Reimers and Gurevych (2019) that such text embeddings might not lead to optimal performance unless fine-tuned specifically for text similarity task.

To benchmark the performance of the BERT models, we implement text vectorization via TF-IDF weighting (Spärck Jones, 1972). TF-IDF is a type of bag-of-words approach, where the numerical representation of the text in vector space is based on a token decomposition of the text, ignoring the sequential nature of the text. The TF-IDF then applies a weighting scheme that puts a higher weight on words that appear frequently in one document, but rarely across documents. The TF-IDF vectorization results in high-dimensional *sparse* vectors, which is in contrast to the *dense* vectors resulting from the BERT models. Such TF-IDF vectorization has proven to be very effective in text similarity tasks, despite its simplicity (compare e.g. Hettich and Pazzani, 2006; Shahmirzadi et al., 2019). We pre-process the texts for TF-IDF as follows: we lower-case the texts first and split the text sequence into separate words, i.e. tokens, while removing stop words and performing stemming of the remaining words.

To investigate the influence of the choice of text representation on the matching results, we evaluate the performance of the transformer models for both CLS token and mean pooling as these are the commonly used embedding extractions in practice (Reimers and Gurevych, 2019), as well as for unigrams and 3-grams in the case of TF-IDF as these represent different levels of granularity of the text (Shahmirzadi et al., 2019).

The matching procedure can be defined as follows. Consider a grant proposal i with $i = 1, ..., N$ in total, while each proposal i is associated with a single text sequence τ_i. Further consider a reviewer j with $j = 1, ..., J$ reviewers in total, while each reviewer is associated with $k = 1, ..., K$ text sequences, resulting in a reviewer-publication text sequence $\rho_{j,k}$. The raw text sequences are then vectorized via vectorization function $\upsilon_m(\cdot)$ depending on the model used

$M \in \{\text{BERT}, \text{SciBERT}, \text{SPECTER2}, \text{TF-IDF}\}$ resulting in the text vectors as:[7]

$$\begin{aligned} &T_i = \upsilon_m(\tau_i) \\ &\text{for} \quad i = 1, ..., N; \quad \forall m \in M \end{aligned} \tag{1}$$

and

$$\begin{aligned} &P_{j,k} = \upsilon_m(\rho_{j,k}) \\ &\text{for} \quad j = 1, ..., J \quad \text{and} \quad k = 1, ..., K; \quad \forall m \in M \end{aligned} \tag{2}$$

Then for all possible pairs of proposals and reviewers' publications, we estimate the text similarity via cosine distance:

$$\hat{\pi}_{i,j,k} = \frac{T_i \cdot P_{j,k}}{\|T_i\| \|P_{j,k}\|}. \tag{3}$$

In order to bring the similarities $\hat{\pi}_{i,j,k}$ onto proposal-reviewer level, for a given proposal-reviewer pair we sort the similarities along the publication level in a decreasing order as

$$\hat{\pi}_{i,j,(1)} \leq \hat{\pi}_{i,j,(2)} \leq \cdots \leq \hat{\pi}_{i,j,(K)} \tag{4}$$

and average the similarities of the 20% most similar publications as follows[8]

$$\hat{\pi}_{i,j} = \frac{1}{K_{20}} \sum_{k=1}^{K_{20}} \hat{\pi}_{i,j,(k)}. \tag{5}$$

Given the average similarities between the pairs of proposals and reviewers $\hat{\pi}_{i,j}$, for each proposal i we rank-order the reviewers j according to their average similarities and select the top R ranked reviewers, with $R \in \{2, 5\}$, to provide a matching recommendation for a subset of suitable reviewers for each proposal as

$$\begin{aligned} &\hat{J}_i^R \in \text{Argmax}_j(\hat{\pi}_{i,j}) \\ &\text{s.t.} \quad |\hat{J}_i^R| = R; \quad \forall R \in \{2, 5\}. \end{aligned} \tag{6}$$

In order to evaluate the quality of the matching recommendation, we compute the Mean Average Precision, i.e. MAP, a common metric for evaluation of recommender systems. (Chen and Liu, 2017). MAP is especially suitable in our case as it takes the ordering information of the proposed matches into account. MAP combines both precision and recall as it approximates the average area under the so-called precision-recall curve (Schütze et al., 2008). In particular, MAP over all grant proposals N can be defined as follows:

$$MAP = \frac{1}{N} \sum_{i=1}^{N} \left(\frac{1}{P_i} \sum_{r=1}^{R} \mu(r) \cdot \frac{n\left(\tilde{J}_i^P \cap \hat{J}_i^R(r)\right)}{r} \right) \tag{7}$$

where P_i is the number of true positive cases, i.e. the matches labelled by the scientific officers,[9] $\mu(\cdot)$ is a so-called relevance function defined as an indicator function equal to 1 if the matched reviewer at rank r is relevant and 0 otherwise, $\tilde{J}_i^P$ is a set of P true recommended reviewers as labelled by the scientific officers, and $\hat{J}_i^R(r)$, denotes indexing of the ordered set of R model recommended reviewers up to the r-th element. Intuitively, MAP equals 1 if the recommended matches correspond exactly to those labelled by the scientific officers for all grant proposals, while it equals 0 if we do not get any correct recommendations. As MAP takes the ordering information of the recommendations into account, even if we on average always do find the true two matches among $R = 5$ recommended ones, yet only at the 4th and 5th rank, the MAP value would correspond to 0.225. Similarly, if we on average find only a single match among the recommended ones, MAP would equal 0.5 if the match was on the first rank, but it would equal only to 0.1 if the match was on the fifth rank. This demonstrates how MAP is a distance sensitive metric and penalizes recommendations at lower ranks. In order to reflect the variability in MAP, we additionally compute the variance across the N proposals.

5 Results

Table 2 below presents the MAP results at $R = 5$, i.e. for the top 5 recommended reviewers, depending on the type of text embedding, number of recent years and the type of input text. We focus on the $R = 5$ case as our main objective is pre-filtering a subset of suitable reviewers, from which the scientific officers can easily choose the two most suitable reviewers. We provide the results for the case of $R = 2$ in Appendix to benchmark the results with

[7] We further suppress the dependence on a specific model m for notational ease.

[8] We tested the influence of this threshold by varying it between 10% and 50% and observed qualitatively similar results.

[9] In our setting, P_i is almost always equal to 2. For a handful of cases $P_i = 1$, if only a single reviewer with at least 10 English publications was available, as well as $P_i = 3$, if scientific officers labelled one extra reviewer as being suitable.

the case of pre-filtering the exact subset of reviewers needed for the final assignment.

Focusing on the first set of results based on the text embeddings via mean pooling, we observe a clear pattern for the BERT models. Regardless of the number of years and the type of input text sequence considered, the MAP is monotonically increasing when switching from BERT to SciBERT and further from SciBERT to SPECTER2. This documents the value added of targeted pre-training of the BERT model on scientific texts and additionally the citation graphs for the matching task based on the text similarity. Interestingly, the TF-IDF model based on 3-grams performs surprisingly well too, in many cases achieving similar performance as the SciBERT model. The unanimously best performance exhibits the SPECTER2 model, which benefits from the pre-training on the citation networks in addition to pre-training on scientific texts.

Looking at the differences based on the varying number of years and text inputs, we uncover additional clear patterns. First, the overall performance of all considered models is only marginally better for the case of including last 10 years of publications instead of 5. As such, additional but less recent data on reviewer's publications do not substantially improve the matching performance on average, although the improvement is greater for the SSH domain as will be discussed below. Second, we observe a sizeable increase in performance, when including abstracts in addition to titles, whereas the performance is *de facto* the same, whether abstracts are included alone or in combination with titles. This pattern is documented for all considered models. Thus, it appears that titles do not contain information that is not also available from the abstract.

Comparing the results based on the mean pooling with those of the CLS tokens, we identify few differences. For the BERT and SciBERT model the performance clearly deteriorates when only the CLS token is used, even more so for SciBERT than for BERT. This provides evidence in favor of text embeddings extraction via mean pooling for the matching task based on text similarity for these particular models. In contrast, the performance of the SPECTER2 model is robust, regardless of the type

Embedding	**Years**	**Text**	**BERT**	**SciBERT**	**SPECTER2**	**TF-IDF**
mean pooling / 3-gram	5	title	0.3117 *(0.1020)*	0.3167 *(0.1054)*	0.3745 *(0.1092)*	0.2316 *(0.0804)*
mean pooling / 3-gram	5	abstract	0.3684 *(0.1110)*	0.3949 *(0.1018)*	0.4518 *(0.1128)*	0.3932 *(0.1115)*
mean pooling / 3-gram	5	title + abstract	0.3653 *(0.1101)*	0.3905 *(0.1012)*	**0.4536** ***(0.1144)***	0.3925 *(0.1101)*
mean pooling / 3-gram	10	title	0.3175 *(0.1054)*	0.3316 *(0.1093)*	0.3842 *(0.1141)*	0.2585 *(0.0893)*
mean pooling / 3-gram	10	abstract	0.3675 *(0.1067)*	0.4205 *(0.1053)*	**0.4687** ***(0.1136)***	0.4000 *(0.1106)*
mean pooling / 3-gram	10	title + abstract	0.3696 *(0.1071)*	0.4184 *(0.1052)*	0.4619 *(0.1161)*	0.4033 *(0.1101)*
CLS token / uni-gram	5	title	0.1937 *(0.0743)*	0.3104 *(0.0968)*	0.3908 *(0.1110)*	0.2305 *(0.0767)*
CLS token / uni-gram	5	abstract	0.2456 *(0.0764)*	0.2001 *(0.0708)*	**0.4554** ***(0.1127)***	0.3792 *(0.1087)*
CLS token / uni-gram	5	title + abstract	0.2719 *(0.0921)*	0.1941 *(0.0668)*	0.4520 *(0.1170)*	0.3692 *(0.1020)*
CLS token / uni-gram	10	title	0.2000 *(0.0807)*	0.3298 *(0.1119)*	0.4034 *(0.1142)*	0.2504 *(0.0842)*
CLS token / uni-gram	10	abstract	0.2718 *(0.0921)*	0.2123 *(0.0716)*	**0.4605** ***(0.1227)***	0.3900 *(0.1077)*
CLS token / uni-gram	10	title + abstract	0.2917 *(0.1047)*	0.1908 *(0.0652)*	0.4576 *(0.1219)*	0.3811 *(0.1036)*

Note: Higher MAP values indicate better performance. Variance displayed below in parentheses.

Table 2: Results on the Mean Average Precision (MAP) at R=5 across models

of embedding. In case of TF-IDF, the results are also quite robust to the type of vectorization as the performance based on uni-grams is similar to that of 3-grams. In terms of the differences based on the number of years and the type of text inputs, we observe the same patterns as for the mean pooling and 3-grams respectively.

We further investigate the overall results by uncovering the heterogeneity with respect to research areas (see Tables 5, 7 and 9 in Appendix).[10] In general, we observe similar patterns in terms of the performance of the considered models. However, we observe a substantial differences in the performance of all the models across the research areas. Most importantly, the results reveal lower performance for the SSH domain in particular. This might be partly due to the under-representation of the SSH domain within the publication data (see Table 3 in Appendix for details). In this regard, for SSH domain we observe a sizeable improvement in the performance of the SPECTER2 model in particular, when including texts of the past 10 years as opposed to 5 years, as can be seen in Tables 5 and 6 in the Appendix. This suggests that including more publication data is valuable for a better matching of reviewers in the SSH domain.[11] Interestingly, TF-IDF performs rather well for the SSH domain, although the contextual information that might be particularly important is not taken into account by this method.

Comparing the overall results with the case of recommending a subset of top 2 most similar reviewers, i.e. $R = 2$, we generally observe the same patterns as for the case of $R = 5$. Based on the conducted analyses, the SPECTER2 model provides the best and most robust performance across different model choices, data inputs, and research areas. Interestingly, a classical TF-IDF model turns out to be also well-performing and robust choice for the matching task based on text similarity.

6 Discussion

In this study, we investigated the value of pre-training BERT models towards scientific domain for the matching task based on text similarity and compared the performance with a classical bag-of-words approach. The results reveal two main findings: First, pre-training on scientific texts and additionally considering the citation networks clearly improves the overlap between the actual and the recommended proposal-reviewer matches. Second, BERT models do not substantially out-perform TF-IDF in the matching tasks, unless both scientific documents and the citation networks are taken into account in the pre-training, i.e. the SPECTER2 model.

These results are in line with the findings of Shahmirzadi et al. (2019), who find the TF-IDF model to perform equally well as other more complex neural models. Nevertheless, the similar performance of the transformer models and TF-IDF is rather surprising, given the large conceptual differences in the text vectorization. One of the possible reasons for this phenomenon might be the fact that extracting raw BERT embeddings is not optimal, unless specifically fine-tuned for the task of text similarity as argued by Reimers and Gurevych (2019). This has also been the approach pursued by Yang et al. (2020) to compare the performance of transformer models for text similarity task in a clinical domain.

Furthermore, the results reveal substantial heterogeneity in the performance across research areas. For all considered models the matching task is the most challenging within the SSH domain. This might be due in part to the diverse type of outputs in these disciplines that are covered less completely in bibliometric databases, for which text similarity might not be the optimal approach. In addition, the large variety of disciplines within the SSH domain might pose another complication for the models considered here, as opposed to domains of MINT and LS, where the proposal and publication texts share more similar characteristics overall. SSH texts sometimes use more generic terminology with less specific keywords than what is found in MINT and LS, and we had hypothesized that methods based on text embeddings would benefit from the incorporation of large contexts, but this did not turn out to be the case. One of the possibilities to overcome this challenge might be an explicit fine-tuning of Siamese networks as suggested by Reimers and Gurevych (2019) on pairs of SSH texts.

Overall, the results presented in this study contribute to a better understanding of the usage of

[10] Results for $R = 2$ by research area are provided in Appendix in Tables 6, 8 and 10.

[11] This improvement might stem from the increased amount of text data itself as well as from the content of the text data which might be more similar across time for SSH than for other research areas, or perhaps that fields within SSH are more distinct from other fields even as the fields themselves change across time.

pre-trained transformer models vs. classical bag-of-words models for a matching task based on text similarity in a scientific domain. The findings of our analyses provide empirical evidence on the suitability and sensitivity of the particular models, data inputs and modelling choices, for matching grant proposals to reviewers - a core task of any research funding agency.

Limitations

The analyses presented in this study have a limited scope. Firstly, the limitations concern the external validity of the results. As our validation dataset focuses on a specific call from a specific funding scheme at the SNSF with a relatively small sample size, it is not assured that the findings are representative for other funding schemes within the SNSF, or broader, for other funding agencies.

Secondly, restricting the data to English texts prevents the assessment of all submitted grant proposals and all potential reviewers. Such restriction further aggravates the imbalances in the availability of text data across research areas, resulting in lower representation of the SSH domain.

Thirdly, our analyses are limited to comparison of BERT models and the TF-IDF model for text vectorization. Therefore, our findings are not representative for newer open-source transformer models such as Llama (Touvron et al., 2023) or Mistral (Jiang et al., 2023), or for other alternative text vectorization methods such as Word2Vec (Mikolov et al., 2013) or GloVe (Pennington et al., 2014), which might potentially out-perform the approaches analyzed here. Additionally, due to the token length limited to 512 tokens for the pre-trained BERT models, the texts are truncated at this threshold, which leads to occasional information loss. We have experimented with truncation from the left and right of the text sequences, which did not change the qualitative conclusions.

Lastly, extracting raw embeddings from the pre-trained BERT models without explicitly fine-tuning the models for the text similarity task might result in sub-optimal performance. Nervetheless, it should provide a reasonable baseline in cases where labelled data is not feasible to collect.

Reproducibility

The code used to conduct this analysis is available at https://github.com/snsf-data/snsf-grant-similarity. Due to the data protection laws, the data used in this study cannot be shared.

Ethics Statement

The results from the NLP algorithm have never been used to directly assign reviewers to grant applications without a validation from the scientific officers. Scientific officers always check and validate the suggested matching and adjust the assignment as necessary.

Acknowledgements

This work could not have been completed without the efforts from our colleagues in Postdoc.Mobility who diligently labeled an entire call from August 2021 with the true best matching reviewers and co-reviewers for each proposal, before considering practical requirements such as conflicts of interest, so that we have a dataset that can be used for testing algorithm variations. This dataset has been invaluable, and we would not have been able to progress on this work without their help. In addition, we would like to thank the four anonymous reviewers for helpful comments and suggestions on a previous version of this manuscript.

Competing Interests

Both authors are employed by the SNSF.

References

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. Scibert: A pretrained language model for scientific text. *arXiv preprint arXiv:1903.10676*.

Kenneth Benoit, Kohei Watanabe, Haiyan Wang, Paul Nulty, Adam Obeng, Stefan Müller, and Akitaka Matsuo. 2018. quanteda: An R package for the quantitative analysis of textual data. *Journal of Open Source Software*, 3(30):774–774.

Dhivya Chandrasekaran and Vijay Mago. 2021. Evolution of semantic similarity—a survey. *ACM Computing Surveys (CSUR)*, 54(2):1–37.

Laurent Charlin and Richard Zemel. 2013. The Toronto paper matching system: an automated paper-reviewer assignment system.

Mingang Chen and Pan Liu. 2017. Performance evaluation of recommender systems. *International Journal of Performability Engineering*, 13(8):1246.

Arman Cohan, Sergey Feldman, Iz Beltagy, Doug Downey, and Daniel S Weld. 2020. Specter: Document-level representation learning using citation-informed transformers. *arXiv preprint arXiv:2004.07180*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Seth Hettich and Michael J Pazzani. 2006. Mining for proposal reviewers: lessons learned at the national science foundation. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 862–871.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Oscar Kjell, Salvatore Giorgi, and H Andrew Schwartz. 2023. The text-package: An R-package for analyzing and visualizing human language using natural language processing and transformers. *Psychological Methods*.

Chandler May, Alex Wang, Shikha Bordia, Samuel R Bowman, and Rachel Rudinger. 2019. On measuring social biases in sentence encoders. *arXiv preprint arXiv:1903.10561*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, 26.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Yifan Qiao, Chenyan Xiong, Zhenghao Liu, and Zhiyuan Liu. 2019. Understanding the behaviors of BERT in ranking. *arXiv preprint arXiv:1904.07531*.

Nils Reimers, Philip Beyer, and Iryna Gurevych. 2016. Task-oriented intrinsic evaluation of semantic textual similarity. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 87–96.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using siamese BERT-networks. *arXiv preprint arXiv:1908.10084*.

Hinrich Schütze, Christopher D Manning, and Prabhakar Raghavan. 2008. *Introduction to information retrieval*, volume 39. Cambridge University Press Cambridge.

Nihar B Shah. 2022. Challenges, experiments, and computational solutions in peer review. *Communications of the ACM*, 65(6):76–87.

Omid Shahmirzadi, Adam Lugowski, and Kenneth Younge. 2019. Text similarity in vector space models: a comparative study. In *2019 18th IEEE international conference on machine learning and applications (ICMLA)*, pages 659–666. IEEE.

Amanpreet Singh, Mike D'Arcy, Arman Cohan, Doug Downey, and Sergey Feldman. 2022. SciRepEval: A multi-format benchmark for scientific document representations. *arXiv preprint arXiv:2211.13308*.

Karen Spärck Jones. 1972. A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation*, 28(1):11–21.

Ivan Stelmakh, Nihar Shah, and Aarti Singh. 2021. PeerReview4All: Fair and accurate reviewer assignment in peer review. *Journal of Machine Learning Research*, 22(163):1–66.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Lewis Tunstall, Leandro Von Werra, and Thomas Wolf. 2022. *Natural language processing with transformers*. " O'Reilly Media, Inc.".

Richard E Turner. 2023. An introduction to transformers. *arXiv preprint arXiv:2304.10557*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2020. HuggingFace's transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 conference on empirical methods in natural language processing: system demonstrations*, pages 38–45.

Letian Wu, Wenyao Zhang, Tengping Jiang, Wankou Yang, Xin Jin, and Wenjun Zeng. 2023. [CLS] token is all you need for zero-shot semantic segmentation. *arXiv preprint arXiv:2304.06212*.

Xi Yang, Xing He, Hansi Zhang, Yinghan Ma, Jiang Bian, Yonghui Wu, et al. 2020. Measurement of semantic textual similarity in clinical texts: comparison of transformer-based models. *JMIR medical informatics*, 8(11):e19735.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. BERTScore: Evaluating text generation with BERT. *arXiv preprint arXiv:1904.09675*.

Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *Proceedings of the IEEE international conference on computer vision*, pages 19–27.

A Appendix

A.1 Descriptive Statistics

Publications	SSH	MINT	LS
5 years	22.6	52.1	44.4
10 years	40.6	96.0	81.2

Table 3: Distribution of the average number of publications per research area

A.2 Model Details

In our analyses, we deploy specifically the following models from the Hugging Face platform (Wolf et al., 2020):

- BERT: google-bert/bert-base-uncased
- SciBERT: allenai/scibert_scivocab_uncased
- SPECTER2: allenai/specter2_base

and use the quanteda (Benoit et al., 2018) implementation for the TF-IDF vectorization.

A.3 Supplementary Results

Embedding	**Years**	**Text**	**BERT**	**SciBERT**	**SPECTER2**	**TF-IDF**
mean pooling / 3-gram	5	title	0.2383 *(0.0882)*	0.2398 *(0.0961)*	0.2945 *(0.1032)*	0.1883 *(0.0763)*
mean pooling / 3-gram	5	abstract	0.2734 *(0.1100)*	0.2938 *(0.0988)*	0.3562 *(0.1188)*	0.3117 *(0.1073)*
mean pooling / 3-gram	5	title + abstract	0.2719 *(0.1104)*	0.2867 *(0.0956)*	**0.3602** ***(0.1181)***	0.3102 *(0.1071)*
mean pooling / 3-gram	10	title	0.2383 *(0.0968)*	0.2531 *(0.1038)*	0.2914 *(0.1070)*	0.2055 *(0.0805)*
mean pooling / 3-gram	10	abstract	0.2734 *(0.1021)*	0.3117 *(0.1026)*	**0.3703** ***(0.1203)***	0.3023 *(0.1076)*
mean pooling / 3-gram	10	title + abstract	0.2758 *(0.1053)*	0.3125 *(0.0995)*	0.3680 *(0.1238)*	0.3094 *(0.1109)*
CLS token / uni-gram	5	title	0.1391 *(0.0676)*	0.2266 *(0.0849)*	0.2938 *(0.0992)*	0.1797 *(0.0715)*
CLS token / uni-gram	5	abstract	0.1727 *(0.0698)*	0.1461 *(0.0622)*	0.3578 *(0.1161)*	0.3078 *(0.1048)*
CLS token / uni-gram	5	title + abstract	0.1938 *(0.0787)*	0.1414 *(0.0585)*	**0.3586** ***(0.1192)***	0.2914 *(0.0953)*
CLS token / uni-gram	10	title	0.1469 *(0.0665)*	0.2508 *(0.0997)*	0.3078 *(0.1064)*	0.1906 *(0.0740)*
CLS token / uni-gram	10	abstract	0.1969 *(0.0810)*	0.1500 *(0.0578)*	0.3648 *(0.1280)*	0.3008 *(0.1069)*
CLS token / uni-gram	10	title + abstract	0.2070 *(0.0932)*	0.1328 *(0.0532)*	**0.3656** ***(0.1265)***	0.2906 *(0.0994)*

Note: Higher MAP values indicate better performance. Variance displayed below in parentheses.

Table 4: Results on the Mean Average Precision (MAP) at R=2 across models

Embedding	Years	Text	BERT	SciBERT	SPECTER2	TF-IDF
mean pooling / 3-gram	5	title	0.2402 *(0.1119)*	0.2273 *(0.0936)*	0.2513 *(0.1147)*	0.1512 *(0.0526)*
mean pooling / 3-gram	5	abstract	0.3080 *(0.1282)*	0.3020 *(0.1060)*	0.3327 *(0.1193)*	0.3533 *(0.1278)*
mean pooling / 3-gram	5	title + abstract	0.2973 *(0.1277)*	0.2983 *(0.1075)*	0.3492 *(0.1319)*	**0.3603** ***(0.1222)***
mean pooling / 3-gram	10	title	0.2440 *(0.0951)*	0.2128 *(0.0809)*	0.2647 *(0.0838)*	0.1840 *(0.0622)*
mean pooling / 3-gram	10	abstract	0.2915 *(0.1095)*	0.3150 *(0.0960)*	**0.4070** ***(0.1270)***	0.3312 *(0.0844)*
mean pooling / 3-gram	10	title + abstract	0.2913 *(0.1057)*	0.3388 *(0.1033)*	0.3908 *(0.1320)*	0.3562 *(0.0910)*
CLS token / uni-gram	5	title	0.1873 *(0.0943)*	0.2213 *(0.0983)*	0.2502 *(0.0968)*	0.1245 *(0.0402)*
CLS token / uni-gram	5	abstract	0.1617 *(0.0621)*	0.2017 *(0.0797)*	**0.3325** ***(0.1135)***	0.3048 *(0.1166)*
CLS token / uni-gram	5	title + abstract	0.1690 *(0.0566)*	0.1735 *(0.0562)*	0.3038 *(0.1164)*	0.2925 *(0.1112)*
CLS token / uni-gram	10	title	0.2098 *(0.0957)*	0.2237 *(0.1040)*	0.2643 *(0.0666)*	0.1643 *(0.0481)*
CLS token / uni-gram	10	abstract	0.2240 *(0.0895)*	0.2060 *(0.0739)*	**0.3775** ***(0.1345)***	0.3328 *(0.0948)*
CLS token / uni-gram	10	title + abstract	0.2318 *(0.0744)*	0.1657 *(0.0694)*	0.3493 *(0.1212)*	0.3140 *(0.0883)*

Note: Higher MAP values indicate better performance. Variance displayed below in parentheses.

Table 5: **Research Area SSH** - Mean Average Precision (MAP) at R=5 across models

Embedding	Years	Text	BERT	SciBERT	SPECTER2	TF-IDF
mean pooling / 3-gram	5	title	0.1900 *(0.1060)*	0.1800 *(0.0945)*	0.2000 *(0.1097)*	0.1100 *(0.0489)*
mean pooling / 3-gram	5	abstract	0.2300 *(0.1271)*	0.1850 *(0.1067)*	0.2500 *(0.1250)*	0.2800 *(0.1266)*
mean pooling / 3-gram	5	title + abstract	0.2250 *(0.1256)*	0.1950 *(0.1053)*	**0.2950** ***(0.1472)***	0.2850 *(0.1148)*
mean pooling / 3-gram	10	title	0.1700 *(0.0700)*	0.1500 *(0.0791)*	0.1850 *(0.0811)*	0.1400 *(0.0565)*
mean pooling / 3-gram	10	abstract	0.2050 *(0.1089)*	0.1850 *(0.0939)*	**0.3050** ***(0.1385)***	0.2150 *(0.0817)*
mean pooling / 3-gram	10	title + abstract	0.2150 *(0.1097)*	0.2350 *(0.0980)*	**0.3050** ***(0.1385)***	0.2350 *(0.0980)*
CLS token / uni-gram	5	title	0.1400 *(0.0820)*	0.1750 *(0.0950)*	0.1900 *(0.0882)*	0.0850 *(0.0398)*
CLS token / uni-gram	5	abstract	0.1150 *(0.0618)*	0.1650 *(0.0832)*	**0.2550** ***(0.1186)***	0.2300 *(0.0965)*
CLS token / uni-gram	5	title + abstract	0.1100 *(0.0514)*	0.1300 *(0.0593)*	0.2350 *(0.1107)*	0.2250 *(0.0950)*
CLS token / uni-gram	10	title	0.1400 *(0.0667)*	0.1750 *(0.0925)*	0.1900 *(0.0678)*	0.1000 *(0.0383)*
CLS token / uni-gram	10	abstract	0.1850 *(0.0862)*	0.1550 *(0.0788)*	**0.3100** ***(0.1341)***	0.2200 *(0.0960)*
CLS token / uni-gram	10	title + abstract	0.1850 *(0.0709)*	0.1250 *(0.0721)*	0.2900 *(0.1157)*	0.2100 *(0.0851)*

Note: Higher MAP values indicate better performance. Variance displayed below in parentheses.

Table 6: **Research Area SSH** - Mean Average Precision (MAP) at R=2 across models

Embedding	Years	Text	BERT	SciBERT	SPECTER2	TF-IDF
mean pooling / 3-gram	5	title	0.2556 *(0.0766)*	0.2711 *(0.0846)*	0.3758 *(0.0896)*	0.2487 *(0.0869)*
mean pooling / 3-gram	5	abstract	0.3530 *(0.1016)*	0.3874 *(0.0931)*	**0.4634** ***(0.1034)***	0.3770 *(0.1003)*
mean pooling / 3-gram	5	title + abstract	0.3493 *(0.0955)*	0.3780 *(0.0917)*	0.4584 *(0.1024)*	0.3794 *(0.1050)*
mean pooling / 3-gram	10	title	0.2502 *(0.0759)*	0.2870 *(0.0805)*	0.3884 *(0.1026)*	0.2675 *(0.0892)*
mean pooling / 3-gram	10	abstract	0.3411 *(0.0960)*	0.4239 *(0.0972)*	**0.4583** ***(0.0998)***	0.3686 *(0.0963)*
mean pooling / 3-gram	10	title + abstract	0.3500 *(0.0964)*	0.4054 *(0.0948)*	0.4460 *(0.0945)*	0.3752 *(0.1023)*
CLS token / uni-gram	5	title	0.1315 *(0.0480)*	0.2673 *(0.0740)*	0.3840 *(0.0910)*	0.2496 *(0.0782)*
CLS token / uni-gram	5	abstract	0.2370 *(0.0768)*	0.1884 *(0.0613)*	0.4353 *(0.1000)*	0.3752 *(0.1024)*
CLS token / uni-gram	5	title + abstract	0.2744 *(0.0912)*	0.1694 *(0.0517)*	**0.4364** ***(0.1010)***	0.3713 *(0.0998)*
CLS token / uni-gram	10	title	0.1446 *(0.0551)*	0.2920 *(0.0901)*	0.4091 *(0.1071)*	0.2558 *(0.0821)*
CLS token / uni-gram	10	abstract	0.2514 *(0.0875)*	0.1970 *(0.0630)*	**0.4430** ***(0.1050)***	0.3619 *(0.1012)*
CLS token / uni-gram	10	title + abstract	0.2761 *(0.1097)*	0.1775 *(0.0510)*	0.4363 *(0.1049)*	0.3622 *(0.0997)*

Note: Higher MAP values indicate better performance. Variance displayed below in parentheses.

Table 7: **Research Area LS** - Mean Average Precision (MAP) at R=5 across models

Embedding	Years	Text	BERT	SciBERT	SPECTER2	TF-IDF
mean pooling / 3-gram	5	title	0.1951 *(0.0671)*	0.2134 *(0.0816)*	0.2927 *(0.0960)*	0.2114 *(0.0851)*
mean pooling / 3-gram	5	abstract	0.2663 *(0.1063)*	0.2967 *(0.0987)*	**0.3638** ***(0.1109)***	0.3008 *(0.1004)*
mean pooling / 3-gram	5	title + abstract	0.2561 *(0.0999)*	0.2846 *(0.0905)*	0.3598 *(0.1057)*	0.3008 *(0.1045)*
mean pooling / 3-gram	10	title	0.1768 *(0.0725)*	0.2195 *(0.0785)*	0.2927 *(0.1022)*	0.2175 *(0.0840)*
mean pooling / 3-gram	10	abstract	0.2622 *(0.0941)*	0.3313 *(0.1050)*	**0.3638** ***(0.1109)***	0.2622 *(0.0869)*
mean pooling / 3-gram	10	title + abstract	0.2642 *(0.0987)*	0.3191 *(0.0976)*	0.3516 *(0.1074)*	0.2785 *(0.0975)*
CLS token / uni-gram	5	title	0.0915 *(0.0443)*	0.1850 *(0.0675)*	0.2785 *(0.0812)*	0.2012 *(0.0724)*
CLS token / uni-gram	5	abstract	0.1585 *(0.0710)*	0.1341 *(0.0546)*	0.3435 *(0.1070)*	0.3069 *(0.1002)*
CLS token / uni-gram	5	title + abstract	0.1911 *(0.0780)*	0.1179 *(0.0423)*	**0.3455** ***(0.1071)***	0.2907 *(0.0895)*
CLS token / uni-gram	10	title	0.1179 *(0.0526)*	0.2195 *(0.0805)*	0.3150 *(0.1043)*	0.2053 *(0.0738)*
CLS token / uni-gram	10	abstract	0.1829 *(0.0769)*	0.1484 *(0.0572)*	0.3415 *(0.1140)*	0.2663 *(0.1001)*
CLS token / uni-gram	10	title + abstract	0.1931 *(0.0971)*	0.1220 *(0.0362)*	**0.3455** ***(0.1163)***	0.2703 *(0.0949)*

Note: Higher MAP values indicate better performance. Variance displayed below in parentheses.

Table 8: **Research Area LS** - Mean Average Precision (MAP) at R=2 across models

Embedding	Years	Text	BERT	SciBERT	SPECTER2	TF-IDF
mean pooling / 3-gram	5	title	0.3830 *(0.1118)*	0.3852 *(0.1189)*	0.4154 *(0.1184)*	0.2448 *(0.0828)*
mean pooling / 3-gram	5	abstract	0.4018 *(0.1120)*	0.4327 *(0.1046)*	0.4827 *(0.1141)*	0.4204 *(0.1155)*
mean pooling / 3-gram	5	title + abstract	0.4018 *(0.1147)*	0.4324 *(0.1035)*	**0.4852** ***(0.1153)***	0.4145 *(0.1109)*
mean pooling / 3-gram	10	title	0.3988 *(0.1227)*	0.4093 *(0.1318)*	0.4213 *(0.1292)*	0.2762 *(0.0975)*
mean pooling / 3-gram	10	abstract	0.4154 *(0.1112)*	0.4536 *(0.1118)*	0.4985 *(0.1199)*	0.4497 *(0.1279)*
mean pooling / 3-gram	10	title + abstract	0.4126 *(0.1138)*	0.4563 *(0.1122)*	**0.4994** ***(0.1270)***	0.4428 *(0.1215)*
CLS token / uni-gram	5	title	0.2478 *(0.0844)*	0.3768 *(0.1080)*	0.4444 *(0.1243)*	0.2505 *(0.0841)*
CLS token / uni-gram	5	abstract	0.2813 *(0.0782)*	0.2094 *(0.0765)*	0.5140 *(0.1156)*	0.4078 *(0.1102)*
CLS token / uni-gram	5	title + abstract	0.3048 *(0.1014)*	0.2217 *(0.0824)*	**0.5153** ***(0.1205)***	0.3935 *(0.0996)*
CLS token / uni-gram	10	title	0.2430 *(0.0937)*	0.3976 *(0.1245)*	0.4459 *(0.1292)*	0.2752 *(0.0960)*
CLS token / uni-gram	10	abstract	0.3051 *(0.0958)*	0.2273 *(0.0785)*	0.5033 *(0.1309)*	0.4329 *(0.1154)*
CLS token / uni-gram	10	title + abstract	0.3251 *(0.1096)*	0.2106 *(0.0757)*	**0.5123** ***(0.1306)***	0.4197 *(0.1100)*

Note: Higher MAP values indicate better performance. Variance displayed below in parentheses.

Table 9: **Research Area MINT** - Mean Average Precision (MAP) at R=5 across models

Embedding	Years	Text	BERT	SciBERT	SPECTER2	TF-IDF
mean pooling / 3-gram	5	title	0.2908 *(0.0959)*	0.2823 *(0.1064)*	0.3282 *(0.1043)*	0.1956 *(0.0766)*
mean pooling / 3-gram	5	abstract	0.2942 *(0.1076)*	0.3282 *(0.0923)*	**0.3861** ***(0.1201)***	0.3316 *(0.1072)*
mean pooling / 3-gram	5	title + abstract	0.3010 *(0.1138)*	0.3197 *(0.0940)*	0.3827 *(0.1184)*	0.3265 *(0.1075)*
mean pooling / 3-gram	10	title	0.3129 *(0.1172)*	0.3163 *(0.1261)*	0.3265 *(0.1161)*	0.2177 *(0.0850)*
mean pooling / 3-gram	10	abstract	0.3061 *(0.1051)*	0.3384 *(0.0983)*	0.3980 *(0.1214)*	0.3656 *(0.1269)*
mean pooling / 3-gram	10	title + abstract	0.3061 *(0.1086)*	0.3333 *(0.1005)*	**0.4031** ***(0.1314)***	0.3605 *(0.1225)*
CLS token / uni-gram	5	title	0.1786 *(0.0796)*	0.2789 *(0.0921)*	0.3418 *(0.1131)*	0.1939 *(0.0786)*
CLS token / uni-gram	5	abstract	0.2041 *(0.0702)*	0.1497 *(0.0622)*	0.4048 *(0.1184)*	0.3350 *(0.1100)*
CLS token / uni-gram	5	title + abstract	0.2245 *(0.0862)*	0.1650 *(0.0715)*	**0.4116** ***(0.1257)***	0.3146 *(0.0994)*
CLS token / uni-gram	10	title	0.1735 *(0.0776)*	0.3027 *(0.1141)*	0.3418 *(0.1165)*	0.2092 *(0.0839)*
CLS token / uni-gram	10	abstract	0.2126 *(0.0834)*	0.1497 *(0.0519)*	0.4031 *(0.1365)*	0.3571 *(0.1113)*
CLS token / uni-gram	10	title + abstract	0.2262 *(0.0979)*	0.1446 *(0.0616)*	**0.4082** ***(0.1362)***	0.3350 *(0.1049)*

Note: Higher MAP values indicate better performance. Variance displayed below in parentheses.

Table 10: **Research Area MINT** - Mean Average Precision (MAP) at R=2 across models