A Checklist Materials

Datasheet: https://github.com/iesl/CSFCube/blob/master/datasheet.md

License: https://github.com/iesl/CSFCube/blob/master/LICENSE.md

B Full Text vs Abstract Annotations

As we note in §4, to examine the effect of annotating the abstract of papers instead of full-text of papers we also conduct a small scale study to examine the scale of differences between the relevance ratings produced each of them. This is done by a single expert annotator annotating relevance based on abstract and full-text separately for 9 query abstracts (3 from each facet) and 5 candidate abstracts each (45 pairs). In making these annotations, queries were picked to ensure all paper-types (§3) were represented and candidates were chosen at random from across all relevance levels. Next abstract based relevances were labelled, following this full-text relevances were labelled. In labelling full-text relevances care was taken to not show abstract based relevances or the abstract text. In making full-text judgements the paper was skimmed for content relevant to the target facet rather than read exhaustively. Every full-text judgement pair took about 5 minutes to complete. Finally, while the presented study isnt intended to be statistically robust we believe it presents a reasonable pilot study in support of the abstract based ratings adopted in our dataset annotation.

C Baseline Methods

The methods we choose to evaluate capture a range of granularities and nature of methods: term based methods, pre-trained model based sentence representations, and whole abstract representation models. Note that some of the methods evaluated are included in our set of methods to construct candidate pools, but as noted in §3 they used unfaceted representations.

`fabs_tfidf`: This is a simple faceted baseline which builds a sparse TF-IDF representation for the sentences corresponding to the query facet in the query abstract. Candidates are represented by their whole abstract TF-IDF representations.

`fabs_bm25`: This represents a baseline identical to `fabs_tfidf` while using the Okapi BM25 weighting scheme.[8]

`fabs_cbow200`: This is a dense bag-of-words representation for the sentences corresponding to the query facet in the query abstract – token embeddings are averaged. As above, candidates are represented by all abstract sentences. The `word2vec` embeddings are trained on $800,000$ abstracts from the S2ORC corpus (§3). We used 200 dimensional word embeddings.

`fabs_tfidfcbow200`: This baseline combines the above baselines where the `word2vec` representations are weighted by TF-IDF weights prior to being averaged.

`SentBERT`: SentBERT [56] represents a sentence level model. In our setup we encode all query facet sentences and all candidate abstract sentences individually with SentBERT, and then use the maximum pairwise sentence cosine similarity between the query and candidate sentences to rank candidates. We evaluate two versions of SentBERT, one fine-tuned only on Natural Language Inference (NLI) datasets as in Reimers and Gurevych [56] and a second model fine-tuned on NLI and a wide variety of paraphrase text. We term these `SentBERT-NLI` and `SentBERT-PP`.[9] We choose to use `SentBERT-PP` given its strong performance on the SciDOCS benchmark [16].[10]

`SimCSE`: SimCSE [22], represents a very recent state of the art sentence similarity model trained in two ways – an unsupervised manner training an encoder to maximise similarity with a a "dropped-out" representation of the same sentence, and a supervised version trained on NLI data. We denote these as `UnSimCSE` and `SuSimCSE`. We use the models `princeton-nlp/unsup-simcse-bert-base-uncased` and `princeton-nlp/sup-simcse-bert-base-uncased` made available through the Hugging Face[11] package.

[8] BM25 implementation: https://github.com/dorianbrown/rank_bm25

[9] we use the `sentence_transformers` package. In this package, `SentBERT-NLI` corresponds to `nli-roberta-base-v2` and `SentBERT-PP` to `paraphrase-TinyBERT-L6-v2`.

[10] Model performances: https://www.sbert.net/docs/pretrained_models.html

[11] https://huggingface.co/

SPECTER: This approach represents a multi-layer transformer based SciBERT model fine-tuned on citation network data [16]. SPECTER operates on titles and the whole abstract of the papers and represents an entirely un-faceted model. Both queries and candidates are represented by their SPECTER embeddings. Note that SPECTER was trained on a corpus of randomly selected scientific documents. We also re-implement and train a version of SPECTER on about 660k computer science paper triples with identical hyper-parameters to SPECTER, we call this in-domain model SPECTER-ID.

In re-ranking the candidate pool for every query, the L2 distance between the query and candidate vectors was used unless specified otherwise.

D Evaluating Citations

Because we included cited papers in our candidate pools, and manually assign relevance judgments for them, this dataset allows us to examine the common assumption that cited paper abstracts will be relevant to a query paper abstract. In this analysis, we find cited papers to pre-dominantly be rated at 0 or 1 levels of relevance, 79% of the times for `background`, 88% of the times for `method`, and 86% of the times for `result`. Given that citations are often considered incidental signals from which to train models and often to evaluate them as well, we believe these observations will have implications for future modeling and evaluation work. We hope future work will use citations with caution, particularly in evaluation setups for tasks similar to the one posed here.

E Potential Applications

A range of important applications rely on computing similarity between scientific texts. Given that our dataset allow evaluation of document similarity methods in general we believe our test collection fills an important gap in the development and benchmarking of methods intended for these applications.

Exploratory Search: Content based search with paradigms such as Query by Example has been considered more suited to exploratory search tasks [41, 17, 45] than keyword based search. Recent work has also seen AI powered literature navigation tools leveraging content based search at varying levels of granularity [20]. We believe our task formulation directly suits this and will allow development of methods intended for these applications.

Patent Search: Hain et al. [25], highlight the case of measuring technological similarities between patents based on the abstracts of patents, and the subsequent employment of this information in mapping patent quality and mapping technological change. Further they also highlight the lack of benchmarks for the measurement of technological similarity between patents [25, Sec 4.3]. While differing in domain we believe our work provides a valuable resource for model development.

Text Matching for Causal Inference: Mozer et al. [48] highlight the importance of text matching for causal inference from observational text data: "matched documents can be used to make unbiased comparisons between groups on external features such as rates of citation". Roberts et al. [57], demonstrate just such a investigation into the gendered biases of citation patterns. The reliance of these analysis on document similarity and matching across a corpus along specific aspects allows our dataset to be of value in developing methods for document matching.

Expert Search: Expert search presents an important application, specially in the contex of peer review, where scientific papers must be matched to experts suited to review it. This often involves computing scientific document similarity [7], a venue where our work proves valuable. In the case of work such as Karimzadehgan et al. [35], which attempts to find experts along all aspects of a scientific paper, our work provides an even stronger resource.

F Extended results

While Section 5 presents $\text{NDCG}_{\%20}$, we additionally report $\text{NDCG}_{\%100}$ in extended results. $\text{NDCG}_{\%100}$ indicates a metric comuted based on the entire pool per query. We note based on Wang et al. [69], that larger pools cause larger values of NDCG, this is observed here. Further model performance at lower values of k, i.e. at the top of the predicted rankings, is still lower indicating

Table 4: Extended test set results for the set of baselines methods. Metrics (R-Precision, Precision and Recall at 20, NDCG$_{\%20}$, NDCG$_{\%100}$) are computed based on a 2-fold cross-validation, represent averages over per-query metrics, and are reported as percentages. SPECTER-ID performance is reported over three training re-runs, the remaining baselines are reported based on a single set of model parameters released by the respective authors.

	background					method				
	RP	P@20	R@20	NDCG$_{\%100}$	NDCG$_{\%20}$	RP	P@20	R@20	NDCG$_{\%100}$	NDCG$_{\%20}$
`fabs_tfidf`	23.35	27.19	45.80	78.14	57.97	09.30	09.83	34.75	57.87	31.20
`fabs_bm25`	20.12	27.81	49.85	79.02	59.39	09.37	11.63	38.29	60.68	34.59
`fabs_cbow200`	19.61	15.94	27.97	67.68	36.56	08.65	08.33	15.69	51.58	21.14
`fabs_tfidfcbow200`	15.92	16.87	27.76	69.77	40.51	07.99	06.01	17.71	51.87	21.70
`SentBERT-PP`	21.24	28.75	46.67	79.14	60.80	10.00	10.83	36.30	59.50	33.40
`SentBERT-NLI`	19.02	25.00	40.13	75.80	54.23	09.11	11.46	02.89	58.52	31.10
`UnSimCSE-BERT`	18.15	23.44	36.05	74.34	51.59	08.86	09.65	27.92	59.21	31.23
`SuSimCSE-BERT`	19.22	22.81	46.75	76.70	55.22	08.58	09.76	29.01	58.54	30.88
SPECTER	**24.81**	**35.31**	**57.45**	82.24	66.70	**11.72**	13.58	40.81	62.77	37.41
SPECTER-ID	24.55 ±1.3	34.17 ±0.5	53.26 ±0.3	**84.31** ±0.8	**69.22** ±1.71	10.53 ±0.3	**16.22** ±1.21	**44.59** ±3.6	**64.63** ±0.4	**42.76** ±0.78
	result					all				
	RP	P@20	R@20	NDCG$_{\%100}$	NDCG$_{\%20}$	RP	P@20	R@20	NDCG$_{\%100}$	NDCG$_{\%20}$
`fabs_tfidf`	11.35	16.28	38.57	66.12	41.24	14.59	17.64	39.69	67.17	43.19
`fabs_bm25`	11.31	20.00	40.40	67.87	45.07	13.50	19.69	42.73	68.97	46.06
`fabs_cbow200`	11.16	10.42	23.44	60.22	30.93	13.08	11.47	22.23	59.64	29.36
`fabs_tfidfcbow200`	10.43	10.69	24.39	60.30	32.79	11.38	11.09	23.13	60.42	31.42
`SentBERT-PP`	13.60	19.83	41.73	71.90	52.35	14.83	19.62	41.41	69.98	48.57
`SentBERT-NLI`	14.23	22.05	46.99	72.13	51.30	14.04	19.42	38.67	68.68	45.39
`UnSimCSE-BERT`	12.00	19.58	38.95	68.44	45.55	12.92	17.41	34.43	67.17	42.59
`SuSimCSE-BERT`	12.37	18.58	39.76	68.78	44.93	13.33	16.95	34.83	67.83	43.45
SPECTER	18.62	23.78	52.72	75.47	56.67	18.29	23.97	50.14	73.30	53.28
SPECTER-ID	**20.09** ±0.92	**27.36** ±0.45	**58.74** ±3.04	**76.49** ±0.41	**60.40** ±1.31	**18.32** ±0.79	**25.74** ±0.22	**52.12** ±1.54	**74.96** ±0.06	**57.22** ±0.70

significant room for improvements. Finally, note that an apt value of k in computing metrics for evaluation will depend on the choice of target application, we believe trends highlighted in our results hold across values of k as per the consistency of relative performance of models on NDCG$_{\%20}$ and NDCG$_{\%100}$.

G Error Analysis

Based on a qualitative examination of per-query ranking performance of `abs_tfidf, SentBERT-PP` and SPECTER-ID we outline a range of factors which lead the baseline models to underperform. We believe the incorporation of modeling to handle these phenomena will lead to improved performance on our dataset. We indicate various error cases through examples of the query facet, false positive top retrievals (FP), or false negative lower ranked retrievals (FN). We mention the query ID for examples in superscripts, use underlines to emphasize important segments, and we only provide the relevant sentences from the abstract in each example due to space constraints.

Salient Aspects: One source of error is the inability of models to identify the most salient aspects for similarity, often expressed only in part of a larger set of facet sentences.

`background Q:` "Many classification problems require decisions among a large number of competing classes."[1791179]

FP: "Several real problems involve the classification of data into categories or classes."

`background Q:` "With the increasing empirical success of distributional models of compositional semantics, it is timely to consider the types of textual logic that such models are capable of capturing. In this paper, we address shortcomings in the ability of current models to capture logical operations such as negation."[1936997]

Nearly all models miss the notion of negation in the above example.

Multiple Aspects: Within a given facet, papers often expressed multiple finer grained aspects, models however often only retrieved based on a single aspect. In the following baseline models often retrieved based on one or the other aspect:

`method Q:` "We present a Few-Shot Relation Classification Dataset (FewRel), ... The relation of each sentence is first recognized by distant supervision methods, and then filtered by crowd-

workers. We adapt the most recent state-of-the-art few-shot learning methods for relation classification and conduct a thorough evaluation of these methods."[53080736]

Domain specific similarities A set of errors also arise from the inability of models to determine similarity between technical concepts. The example represents an inability to rate "stacking", "ensemble strategy", and "bagging" as similar.

`result Q:` "Using a public corpus, we show that stacking can improve the efficiency of automatically induced anti-spam filters, . . . "[3264891]

FN: "The experiments on standard WEBSPAM-UK2006 benchmark showed that the ensemble strategy can improve the web spam detection performance effectively."

FN: "We evaluate the classifier performances and find that BAGGING performs the best. . . . our method may be an excellent means to classify spam emails"

Mechanistic similarities: Nearly all methods perform poorly in the case of determining mechanistic similarity in `method` facets. This often relies on determining similarity across a sequence of actions. Baseline models failed to align steps [1] and [2] across abstracts below.

`method Q:` "Using an annotated set of"factual"and"feeling"debate forum posts, [1]we extract patterns that are highly correlated with factual and emotional arguments, and [2]then apply a bootstrapping methodology to find new patterns in a larger pool of unannotated forum posts."[10010426]

FN: "[1]High-precision classifiers label unannotated data to automatically create a large training set, which is then given to an extraction pattern learning algorithm. [2]The learned patterns are then used to identify more subjective sentences."

Context dependence of facets: Faceted similarities as labelled here often also show context dependence on other facets. This is notable in the case of `result` queries. Given that one major guideline for result similarity in our dataset are if "the same finding or conclusion" is found, being able to determine context similarity is important.

`result Q:` "...Subsequently, lexical cue proportions, predicted certainty, as well as their time course characteristics are used to compute veracity for each rumor tweet .... Evaluated on the data portion for which hand-labeled examples were available, it achieves .74 F1-score on identifying rumor resolving tweets and .76 F1-score on predicting if a rumor is resolved as true or false."[5052952]

FN: "In this study, we propose a novel approach to capture the temporal characteristics of these features based on the time series of rumor's lifecycle, for which time series modeling technique is applied to incorporate various social context information. Our experiments using the events in two microblog datasets confirm that the method outperforms state-of-the-art rumor detection approaches by large margins."

In these examples, determining that the higher level result of time series information being important for identifying rumour tweets relies on modeling method similarity. We believe approaches which improve upon `method` similarity, will likely benefit overall performance on other facets as well.

Qualitative result statements: Finally, we also note that `result` queries which summarize qualitative findings often perform poorer, often requiring broader context and often lacking in term overlaps which may otherwise easily indicate relevance.

`result Q:` "Experiments with several Reddit forums show that style is a better indicator of community identity than topic, even for communities organized around specific topics. Further, there is a positive correlation between the community reception to a contribution and the style similarity to that community, but not so for topic similarity."[11629674]

H Potential training data sources

Given these challenges, we also highlight specific other sources of data that future work may exploit to train models to overcome these problems:

Domain specific paraphrase datasets: Given the reasonably strong performance of the `SentBERT-PP` model, fine-tuned on paraphrase datasets, we believe other domain specific paraphrase

datasets have the potential to be useful for the proposed task. An example is PARADE [26] which presents a dataset of computer science paraphrase pairs.

Selecting informative citation examples: Appendix D presents an analysis of citation data and indicates how only a part of this data contains fine-grained facet similarities. An potential approach to selecting more informative citation examples might involve model dependent training data subset selection approaches such as that proposed in Antonello et al. [5].

Co-citations data: Given that the proposed task relies on capturing fine-grained similarities, co-citations examples in the full-text of papers – papers cited in a narrow context (such as a sentence or paragraph), also promise to contain finer grained similarities likely to help train better models [36]. Use of these examples is specially promoted by existence of parsed full-text data in in the S2ORC corpus.