Optimizing singular value based similarity measures for document similarity comparisons

Jarkko Lagus
IPRally Technologies Oy and
University of Helsinki
jarkko@iprally.com

Arto Klami
Department of Computer Science
University of Helsinki
arto.klami@helsinki.fi

Abstract

The similarity of documents is typically computed using fairly simple similarity measures, such as mean or maximum pooling of word representations followed by vector cosine similarity. This results in fast computation but compared to second-order or matrix-based similarity measures loses information. In this work, we investigate the value of matrix similarity measures for document similarity comparison in full-length patent retrieval tasks and introduce two new metrics motivated by the Schatten p-norm. The new similarity measures are based on singular values and involve learnable parameters to be optimized for a given evaluation task. We show that tuning the similarity measures for a specific task improves the similarity comparison accuracy.

1 Introduction

1.1 Document representations and similarity

For natural language processing tasks, we typically represent words and documents as numerical vectors, since they allow mathematically simple comparisons (e.g. similarity between two documents) and are space-efficient. Modern vector representation methods are highly informative for individual words and even long documents can be represented as relatively low-dimensional vectors. Already simple mean pooling can work well in practice (Conneau et al., 2018), and further developments such as smart weighting schemes (Arora et al., 2017; Gupta et al., 2020), directly learned document vector representations (Le and Mikolov, 2014; Chen, 2017), and especially the contextual embeddings and transformer models (Vaswani et al., 2017; Devlin et al., 2018) have pushed the limits of what one can encode into a vector. Transformers, however, have often very high computational cost (Sharir et al., 2020) and simpler methods and better similarity measures based on static word representations still have their place in many applications.

We step outside such vector-shaped representations and directly work with a full matrix $A \in R^{n \times d}$ that stores d-dimensional representations for n words appearing in the document. We explore the value of covariance pooling and singular value (SV) based similarity measures in patent similarity comparison tasks, and show that in the case of static embeddings, these similarity measures outperform mean vector representations in full document comparison tasks.

The key contribution of this work is the introduction of new matrix similarity measures for document similarity. We explain how submultiplicative norms can be converted into a metric resembling cosine similarity, providing a family of similarity measures building on the Schatten p-norm (later just p-norm) computed using SVs of covariance pooling. We then introduce new similarity measures that are based on the same SVs but map them to similarity scores in a more flexible manner. The new similarity measures have learnable parameters that are tuned for a specific end task and hence can learn to represent relevant information better.

1.2 Matrix metrics

We define a matrix similarity measure to be a function $f(A, B) \in R$ that assigns similarity score for document matrices $A \in R^{n \times d}$ and $B \in R^{m \times d}$, where d is the dimensionality of the word embedding vectors (here $d = 300$), n is the amount of tokens in the document A, and m is the amount of tokens in document B.

The similarity between matrices can be defined in multiple ways. The most straightforward ones – such as Word mover's distance (Kusner et al., 2015) (also known as Bures-Wasserstein distance (Bhatia et al., 2019)) or pairwise comparison of all possible word pairs in a matrix – can be directly applied on matrices with an arbitrary number of rows, and hence for documents of arbitrary lengths. Some other similarity measures assume A and B

to be the same shape. To apply those for document comparisons, we need to first preprocess the document matrices suitably; we here call this step *pooling*. The simplest pooling approach is *padding* the shorter document with suitably many rows of zeros, whereas a more general approach is to use *covariance pooling* where we use $A^TA \in R^{d\times d}$ and $B^TB \in R^{d\times d}$ as the inputs for the similarity measure. Covariance pooling has been shown to have beneficial properties as a document representation (Torki, 2018; Lagus et al., 2019). As d is often large, for the smaller size we can use *SVD pooling* where only k leading singular vectors of the covariance representation are used. This can have a regularizing effect in addition to lowering memory and computational costs (Lagus et al., 2019).

1.3 Patent retrieval as context

We evaluate the measures in the context of patent applications, as an example domain with long but structured documents. Tools for handling patent documents are in high demand due to the high labor cost of manual inspection. This is especially the case for the invalidity search stage, aiming to find relevant patents that could possibly cause issues with e.g. patent infringement, or lead to delays or rejection of the application. Over the years, there has been lots of research on how to automate different parts of the process (Balsmeier et al., 2018; Aristodemou and Tietze, 2018) and on end-to-end solutions (Gao et al., 2022) for specific tasks. In addition to trying to solve specific tasks, there have been efforts toward creating patent-text-specific language models (Lee and Hsiang, 2020; Bekamiri et al., 2021). The patent domain is ideal for exploring alternative similarity measures as the documents are often tens of pages long and better methods are needed to use the full information.

2 New similarity measures

This section introduces our technical contributions. We first explain how submultiplicative matrix norms can be used for deriving a similarity measure between two matrices and provide a family of measures building on the p-norm, computed using SVs of the covariance pooling of document matrices. We then introduce a family of more expressive matrix similarity measures, replacing the matrix norm with alternative functions of the SVs. The new measures have learnable parameters that can be fine-tuned for a given task.

2.1 From matrix norm to similarity measure

Any submultiplicative matrix norm $\|A\|$ satisfying $\|AB\| \leq \|A\|\|B\|$ can be used as a basis for a normalized similarity measure between matrices A and B. If we denote the norm (or norm-like function) with $S(\cdot)$, we get the general formula

$$D(A, B, S(\cdot)) := \frac{S(A^TB)}{S(A^TA)^{1/2}S(B^TB)^{1/2}}. \quad (1)$$

This measure is a natural extension to the standard cosine similarity between vectors. Due to submultiplicativity, it is always within the range $[-1, 1]$. Even though the measure will not in general be a proper metric, we will have higher similarity when A and B are similar in terms of the norm and can use it for similarity comparisons.

We build on a particular family of submultiplicative norms called Schatten p-norms, defined as

$$S_p(A) := \left(\sum_n s_n^p(A)\right)^{1/p}, \quad (2)$$

where $p \in [1, \infty)$ and $s_n(A)$ is the nth SV of the matrix A in descending order. The normalized similarity measure can then be expressed as $D(A, B, S_p(\cdot))$ in the general notation of Eq. (1). This family generalizes several well-known norms: for $p = 2$ we get the Frobenius norm, for $p = 1$ it corresponds to the trace norm, and for $p = \infty$ we get the operator norm. Lagus et al. (2019) presented the similarity measure of Eq. (1) in the specific context of the Frobenius form, but here we consider the general formulation for arbitrary norms and norm-like functions.

For $p \in (0, 1)$ the p-norm is a quasinorm as it does not fulfill the triangle inequality, but we still have $D(A, B, S_p(\cdot)) \in [-1, 1]$ and hence get a normalized similarity measure. The p-quasinorm has gained traction in other matrix applications such as low-rank matrix recovery (Zhang et al., 2019) and image denoising (Xie et al., 2016).

2.2 Learnable similarity measures

The measure (1) depends on the norm. Instead of assuming a specific norm in advance, we propose using a slightly more flexible parametric family of norms. We can then optimize the parameters of the norm directly for a task where the distance measure is used. The p-norm (2) itself has the parameter p which can be learned to maximize a task performance, such as retrieval accuracy.

For more flexibility, we propose extensions of the p-norm that involve additional control parameters. We start from the observation that the p-norm is based on SVs, and construct two alternatives that use SVs as inputs.

The simplest extension

$$S_{w,p}(A) := \left(\sum_n \big(w_n s_n(A)\big)^p \right)^{1/p} \quad (3)$$

weights each SV independently but otherwise retains the functional form of the p-norm. This generalization is still a norm, since for any matrix A, we can always find matrix A' where $s_i(A') = w_i s_i(A)$. One motivation for this norm is the observation of Arora et al. (2017) that removing the direction of the largest singular vector reduces the effect of the most common words that are often uninformative. For $p = 1$ (denoted as $S_{w,1}(\cdot)$ later on) we obtain simple weighting as special case of the more general weighting. Alternatively, we can interpret the weights w_n as a form of an attention mechanism.

As a still more flexible alternative, we consider directly mapping the SVs of $A^T B$ to the similarity with a flexible model. We can then include the normalization within the measure itself, and hence get directly a replacement for Eq. (1). For this, we use a small neural network

$$\mathrm{D}_{NN}(A, B) = T(R(R(s(A^T B)W_1)W_2)W_3),$$

where $R(\cdot)$ is a the rectified linear unit activation function and the layer weights

$$W_1 \in R^{d \times 500},$$
$$W_2 \in R^{500 \times 500}, \text{and}$$
$$W_3 \in R^{500 \times 1}.$$

Finally, the hyperbolic tangent $T(\cdot)$ at the end ensures the output is normalized between $[-1, 1]$. Each layer has also a bias term of suitable size, which is omitted here for conciseness. The network architecture could be further tuned by standard architecture search and hence this architecture is to be seen as one practical example of the more general approach.

3 Experiments

We evaluate the proposed similarity measures in the context of patents. When patent examiners evaluate the novelty of a patent application, there are different kind of prior art that is to be considered. The *X citations* are prior work that can alone lead to a rejection, while the *A citations* describe the state of the art, but are not immediate reasons for rejection. Differentiating between these categories can be useful, for example, in retrieval tasks where we want to rank the patents by their relevance to the original document. If we know the relative ordering of each citation class, we can reorder the search results to highlight the most relevant documents, i.e. in this case *X*s before *A*s.

Patents themselves consist of two main parts, *Claims* and *Description*, where the Claims part describes the actual claims that are being made and the description part is a more free-form description of the invention overall. For this reason, the Claims part is usually much shorter and less noisy than the Description part, while the Description part is more thorough and thus contains more fine-grained information. We evaluate the similarity measures for both cases to provide two parallel sets of results.

3.1 Data and evaluation

Documents and encoding The dataset consist of 3,500 full-length patent applications acquired from the United States Patent and Trademark Office, with average document length being 37,754 characters for the Descriptions and 1,907 characters for the Claims. We encode the patent documents using English 300-dimensional `fastText` embeddings (Joulin et al., 2016) and form the covariance matrices of dimensionality 300×300 of each document as the representation.

Training For the models that require learning the parameters, we use `PyTorch` (Paszke et al., 2019) library to do gradient-based optimization using 2,000 samples as the training set and 500 samples as the validation set. We use triplet loss as the loss function setting one of the models as the distance function and the margin (chosen using hyperparameter optimization) to 0.5. The loss for one instance for the measure in Eq. (1) is then

$$\max(D(A, P, S(\cdot)) - D(A, N, S(\cdot)) + 0.5, 0),$$

and for the neural network model it is

$$\max(\mathrm{D}_{NN}(A, P) - \mathrm{D}_{NN}(A, N) + 0.5, 0),$$

where A is the encoded original document, P is the encoded X citation (positive sample), N is the encoded A citation (negative sample), and $S(\cdot)$ is a norm-like measure. Optimization is terminated once the result on the validation set decreases for three consecutive evaluations.

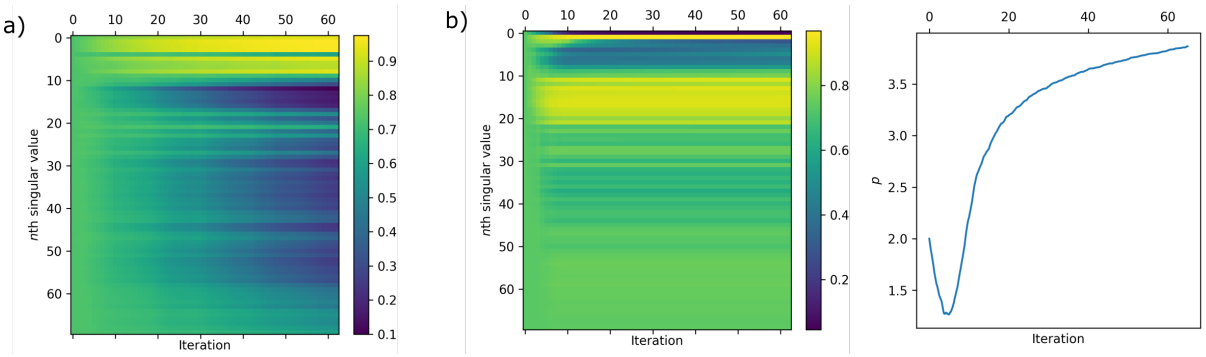

Figure 1: a) Development of singular value weights as a function of iterations for the model $S_{w,1}$. b) Development of the weights and p for the model $S_{w,p}$. Only first 70 out of 300 weights are shown; the rest are effectively zero.

Dataset	Mean	$S_{0.1}$	$S_{0.2}$	$S_{0.5}$	$S_{1.0}$	$S_{1.5}$	$S_{2.0}$	$S_{5.0}$	S_{∞}	S_{opt}	$S_{w,1}$	$S_{w,p}$	D_{NN}
Claims	0.566	0.593	0.601	0.580	0.594	0.577	0.558	0.545	0.545	0.603	0.588	0.589	0.642
Descr.	0.553	0.549	0.558	0.573	0.520	0.504	0.496	0.482	0.482	0.574	0.525	0.574	0.652

Table 1: Numerical results. *Mean* shows the baseline of mean vector with cosine similarity. Free-form neural network model D_{NN} is clearly the best for both tasks.

Evaluation Finally we evaluate the trained model using a test set of 1,000 triplets, measuring the distance from the anchor to both positive and negative samples and counting how often the positive sample is closer to the anchor than the negative sample, i.e. the X citation ranks higher than the A citation. As the baseline, we use the standard mean vector combined with cosine similarity.

3.2 Results

Results are reported in Table 1. We first inspect the accuracy using standard p-norm by grid search over p. The main observation is that small values of p are the best, so that $p = 1$ is the best of the proper norms in both cases and the highest overall accuracy is obtained with quasinorms with $p < 1$. The best p clearly outperforms the baseline of mean vector and cosine similarity (*Mean*); for Claims we improve from 0.566 to 0.601 with $p = 0.2$ and for Descriptions from 0.553 to 0.573 with $p = 0.5$. Large p are clearly worse and all $p > 3$ are effectively equivalent to $p = \infty$.

Instead of evaluating the metric for a range of p, we can directly optimize over p. For both cases, the solution (S_{opt}), slightly improves from the one chosen amongst the grid of alternatives as expected, with optimal values of $p = 0.884$ for Claims and $p = 0.327$ for Descriptions. One technical aspect we note is that when $p \in (0, 1)$ the function is non-convex (Shang et al., 2020) and can have multiple local optima within this range, but we did not observe this to be a problem in practice.

The weighted extension of p-norm of (3) is denoted here by $S_{w,p}$. Figure 1 (a) illustrates the learned weights (as function of iteration) for fixed $p = 1$, demonstrating how the measure assigns more weight for the first 10 or so SVs. Figure 1 (b) illustrates the behavior of the weights and p when optimized jointly, and reveals quite different phenomena: Instead of small p it is now better to use large p and down-weight many of the early singular vectors. For both Claims and Descriptions, the weighted variant $S_{w,p}$ outperforms the mean baseline, but does not provide an improvement over S_{opt} and for Claims it remains worse. One advantage of these measures is that – as seen here – the similarity measures only depend on a fairly small number of SVs; it is enough to compute some tens of the SVs rather than all 300.

The still more flexible neural network measure D_{NN} works well, reaching the highest accuracy for both Claims and Descriptions, with substantial improvement also over S_{opt}. This verifies that SVs of $A^T B$ can be used as the basis for accurately measuring similarity between documents. Importantly, we have high accuracy also for the full-length documents (Descriptions) that are challenging for all other similarity measures.

4 Conclusions

We set out to investigate how similarity measures based on matrix norms work in document similarity comparisons in the context of patent retrieval. We focused on similarity measures based on singular

values of the inner product of the two document matrices, motivated by the p-norm. Our main contribution was introducing new parametric similarity measures that build on the same singular values but are fine-tuned for the specific task at hand, and we showed how a direct neural network mapping the singular values to a distance outperforms both standard mean representation as well as our attempts of more constrained and interpretable measures. In this work we did not fine-tune the neural network architecture to maximize the accuracy but rather used a generic small network, but for practical use the network architecture could be tuned to further improve the accuracy.

While the work was done in the context of static embeddings and patent data, the applicability is not limited to these. Likely any full-document comparison task can benefit from richer representations and the contextual embedding models should enhance the results even further.

Acknowledgments

This work was supported by the Academy of Finland Flagship programme: Finnish Center for Artificial Intelligence FCAI. We thank Janne Sinkkonen for the initial idea and insightful discussions related to the use of Schatten p norm in document comparison.

References

Leonidas Aristodemou and Frank Tietze. 2018. The state-of-the-art on intellectual property analytics (ipa): A literature review on artificial intelligence, machine learning and deep learning methods for analysing intellectual property (ip) data. *World Patent Information*, 55:37–51.

Sanjeev Arora, Yingyu Liang, and Tengyu Ma. 2017. A simple but tough-to-beat baseline for sentence embeddings. In *International conference on learning representations*.

Benjamin Balsmeier, Mohamad Assaf, Tyler Chesebro, Gabe Fierro, Kevin Johnson, Scott Johnson, Guan-Cheng Li, Sonja Lück, Doug O'Reagan, Bill Yeh, et al. 2018. Machine learning and natural language processing on the patent corpus: Data, tools, and new measures. *Journal of Economics & Management Strategy*, 27(3):535–553.

Hamid Bekamiri, Daniel S Hain, and Roman Jurowetzki. 2021. Patentsberta: A deep nlp based hybrid model for patent distance and classification using augmented sbert. *arXiv preprint arXiv:2103.11933*.

Rajendra Bhatia, Tanvi Jain, and Yongdo Lim. 2019. On the bures–wasserstein distance between positive definite matrices. *Expositiones Mathematicae*, 37(2):165–191.

Minmin Chen. 2017. Efficient vector representation for documents through corruption. *arXiv preprint arXiv:1707.02377*.

Alexis Conneau, German Kruszewski, Guillaume Lample, Loïc Barrault, and Marco Baroni. 2018. What you can cram into a single vector: Probing sentence embeddings for linguistic properties. *arXiv preprint arXiv:1805.01070*.

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*.

Xiaochen Gao, Zhaoyi Hou, Yifei Ning, Kewen Zhao, Beilei He, Jingbo Shang, and Vish Krishnan. 2022. Towards comprehensive patent approval predictions: Beyond traditional document classification. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 349–372.

Vivek Gupta, Ankit Saw, Pegah Nokhiz, Praneeth Netrapalli, Piyush Rai, and Partha Talukdar. 2020. P-sif: Document embeddings using partition averaging. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 7863–7870.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2016. Bag of tricks for efficient text classification. *arXiv preprint arXiv:1607.01759*.

Matt Kusner, Yu Sun, Nicholas Kolkin, and Kilian Weinberger. 2015. From word embeddings to document distances. In *International conference on machine learning*, pages 957–966. PMLR.

Jarkko Lagus, Janne Sinkkonen, Arto Klami, et al. 2019. Low-rank approximations of second-order document representations. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*. ACL.

Quoc Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *International conference on machine learning*, pages 1188–1196. PMLR.

Jieh-Sheng Lee and Jieh Hsiang. 2020. Patent classification by fine-tuning bert language model. *World Patent Information*, 61:101965.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. PyTorch:

An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.

Fanhua Shang, Yuanyuan Liu, Fanjie Shang, Hongying Liu, Lin Kong, and Licheng Jiao. 2020. A unified scalable equivalent formulation for schatten quasi-norms. *Mathematics*, 8(8):1325.

Or Sharir, Barak Peleg, and Yoav Shoham. 2020. The cost of training nlp models: A concise overview. *arXiv preprint arXiv:2004.08900*.

Marwan Torki. 2018. A document descriptor using covariance of word vectors. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 527–532.

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.

Yuan Xie, Shuhang Gu, Yan Liu, Wangmeng Zuo, Wensheng Zhang, and Lei Zhang. 2016. Weighted schatten p-norm minimization for image denoising and background subtraction. *IEEE transactions on image processing*, 25(10):4842–4857.

Hengmin Zhang, Jianjun Qian, Bob Zhang, Jian Yang, Chen Gong, and Yang Wei. 2019. Low-rank matrix recovery via modified schatten-p norm minimization with convergence guarantees. *IEEE Transactions on Image Processing*, 29:3132–3142.