Where to Submit? Helping Researchers to Choose the Right Venue

Konstantin Kobs
University of Würzburg
kobs@informatik.uni-wuerzburg.de

Tobias Koopmann
University of Würzburg
koopmann@informatik.uni-wuerzburg.de

Albin Zehe
University of Würzburg
zehe@informatik.uni-wuerzburg.de

David Fernes
University of Würzburg
david.fernes@stud-mail.uni-wuerzburg.de

Philipp Krop
University of Würzburg
philipp.krop@uni-wuerzburg.de

Andreas Hotho
University of Würzburg
hotho@informatik.uni-wuerzburg.de

Abstract

Whenever researchers write a paper, the same question occurs: "Where to submit?" In this work, we introduce WTS, an open and interpretable NLP system that recommends conferences and journals to researchers based on the title, abstract, and/or keywords of a given paper. We adapt the TextCNN architecture and automatically analyze its predictions using the Integrated Gradients method to highlight words and phrases that led to the recommendation of a scientific venue. We train and test our method on publications from the fields of artificial intelligence (AI) and medicine, both derived from the Semantic Scholar dataset. WTS achieves an Accuracy@5 of approximately 83 % for AI papers and 95 % in the field of medicine. It is open source[1] and available for testing on https://wheretosubmit.ml.

1 Introduction

When choosing a scientific conference or journal (in the following called *venue*) to submit a manuscript, researchers consider several factors. While factors such as the venue's impact, time, or location are important, the main factor is the manuscript's thematic fit to the conference. This can be ensured by inspecting the Call for Papers (e.g. this paper fits into EMNLP as it is an "interesting application nugget") or by analyzing previously published papers at the given conference. Given the growing number of conferences (e.g., the exponential growth of computer science publications indicates more and/or larger conferences[2]), the second approach has become harder than ever, especially for novice researchers, or even senior researchers wanting to publish in a new domain. Finding a thematically fitting venue for a manuscript therefore is a time consuming task.
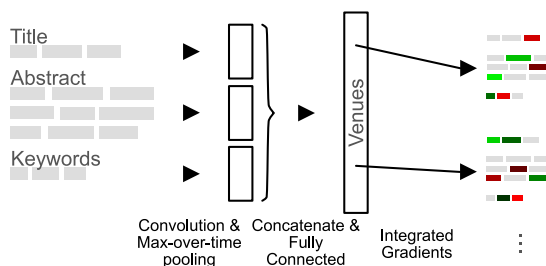

Figure 1: Overview of WTS. Title, abstract, and keywords are processed separately by a convolution layer and max-over-time pooling. The output vectors are concatenated and fed through two fully connected layers that predict fitting venues. For the top five entries, we calculate the important words and phrases using the Integrated Gradient method (Sundararajan et al., 2017).

In this work, we try to simplify this process by introducing "Where to Submit" (WTS), an NLP system based on a convolutional neural network that recommends academic venues given the title, abstract, and/or keywords of a planned publication. The system is trained on previously published manuscripts. To understand the system's choice of recommending a specific venue, WTS analyzes the words and phrases that had the highest impact on a recommendation. A researcher can then use the list of recommended venues as a starting point to find the best fitting venue based on other factors such as the Call for Papers, rank, or deadline. Our main contributions are: (1) We describe an effective method that can recommend scientific venues based on a paper's title, abstract, and keywords. (2) We incorporate an Explainable AI method into our system to give feedback on why a certain conference or journal was recommended. (3) We evaluate our approach on two datasets from two different research areas to show its general applicability. (4) We make WTS available as a web service for everyone to use.

---

[1] https://github.com/konstantinkobs/wts

[2] As visualized on https://dblp.uni-trier.de/statistics/recordsindblp.

2 Related Work

There exist several online services that recommend venues based on the contents of a publication, but all of them are lacking in some ways:

1. Most of them only recommend journals, not conferences (e.g. Elsevier; Journal Guide; Springer; Wiley; Enago; Edanz; Manuscript Matcher; Journal/Author Name Estimator; SJFinder). Especially in the fields of Computer Science and AI, most work is published on conferences (Vrettas and Sanderson, 2015), making this a severe drawback for AI researchers.

2. Most of the services are commercially motivated (Elsevier; IEEE; Springer; Wiley; Enago; Edanz; Manuscript Matcher). Publishers and companies provide them to promote their own portfolio or other services. Thus, they diminish the variety of the recommendations by only considering their own journals.

3. Many of the services are black boxes without any information on how they perform their recommendations. There are a few exceptions to this: Journal/Author Name Estimator uses the open source search engine software Lucene to find the 50 most similar papers according to the Lucene index and recommends the journals that occur most often in this set (Schuemie and Kors, 2008). Elsevier extracts noun phrases from the paper and matches these with a database using the Okapi BM25 algorithm (Kang et al., 2015; Robertson, 1990).

4. None of the provided services explain why a specific venue was chosen. Only very recently, recommending *conferences* based on authors, abstracts, and keywords became a new research area (Iana et al., 2019). However, the authors approach a more general setting that includes conferences from a wide variety of fields. They also incorporate author information into the recommendation and do not provide an explanation to the user why a given conference was recommended. With WTS, we introduce an open and explainable system that recommends both journals and conferences.

3 Task and Methodology

Now we describe the task, our proposed method WTS, and the baselines we compare it to.

3.1 Task Definition

Given a title, abstract, and keywords of a publication, we aim to predict the venue where the paper was published. We interpret the classification task

as a ranking task by ordering the potential venues according to their score in the model output and use metrics that assess the ranking performance.

3.2 Approaches

In the following, we describe the applied baseline methods as well as our own approach.

Random Baseline The simplest baseline is to always predict venues in a uniformly random order. As this will yield variances in prediction quality, we report the expected value for each metric.

Majority Baseline The majority baseline orders venues by the number of publications in the training set in descending order. Assuming stratified sampling, common venues are ranked higher.

Logistic Regression Baseline For this baseline, we tokenize the title, abstract, and keywords. From all tokens, we create a term frequency vector and train a multi-class logistic regression. The venues are then sorted in decreasing order based on the model output. For implementation, we use sklearn's methods for vectorization and logistic regression (Pedregosa et al., 2011).

Iana et al. We also compare our method to one of the approaches outlined in (Iana et al., 2019). For better comparison, we use their best performing approach (according to Recall@10) that does not incorporate any third-party information (called "Ensemble TF-IDF & word2vec plus CNN (10)"). A logistic regression combines two classifiers: (1) Concatenating all corresponding abstracts of a venue, creating one TF-IDF representation and ranking venues using their distance to the provided abstract representation and (2) classifying abstracts using TextCNN, a convolutional neural network (CNN) for text classification (Kim, 2014).

WTS (Ours) Our model is based on the TextCNN and implemented in PyTorch (Paszke et al., 2019). Our network's structure is shown in Figure 1. In contrast to Iana et al., we also provide the network with title and keyword information. We lowercase and embed each word in the title, abstract, and keywords using Word2Vec (Mikolov et al., 2013), trained on the abstracts and titles of the respective dataset. This creates three two-dimensional inputs for the model. Each input is then processed through a convolution layer with potentially multiple filter sizes and max-over-time pooling, which maps the processed inputs to a fixed

Metric	AI	Medicine
Publications	245 573	2 924 609
Venues	78	78
Avg Title length	9.24	13.29
Avg Abstract length	153.65	185.18
Avg used Keywords	8.73	12.33
# total Keywords	32 139	209 525
Min. publication count	63	20 959
Mean publication count	3148.37	37 494.99
Standard Deviation	4102.96	28 170.86
Median publication count	1597.5	27 734.5
Max. publications count	21 122	201 469

Table 1: Comparison of the subsets for AI and medicine venues from Semantic Scholar.

	Hyperparam.	AI	Medicine
Title	# Filters*	89	265
	Filter Sizes*	[7]	[7]
	Input Length	75	75
Abstract	# Filters*	272	265
	Filter Sizes*	[1]	[1]
	Input Length	200	200
Keywords	# Filters*	219	265
	Filter Sizes*	[7]	[7]
	Input Length	25	25
General	Learning Rate*	0.083	0.041
	Dropout*	0.42	0.14
	Early Stopping Patience	10	10
	Optimizer*	SGD	SGD

Table 2: Hyperparameters for our model. Parameters marked with * were determined by random search.

size. The resulting vectors are concatenated and fed through two feed-forward layers that map to a vector representing the venues. Training with categorical cross entropy leads to higher outputs for more likely venues. Dropout (Srivastava et al., 2014) and batch normalization (Ioffe and Szegedy, 2015) are used for regularization.

4 Datasets

For training and testing, we extract all publications from the Semantic Scholar dataset (Ammar et al., 2018)[3] that were published in the research fields *artificial intelligence* (AI) and *medicine*.[4]

A publication is considered to be an AI paper if it was published in one of the scientific venues given in (Kersting et al., 2019). We manually match them as closely as possible to the Semantic Scholar venues. This procedure leads to 77 distinct venues. We also add a class called "non-AI" consisting of 20 000 publications from other fields, to let the model learn the difference between AI and non-AI venues, resulting in 78 classes for this dataset.

In the field of medicine, we only use publications from Semantic Scholar that originate from Medline, a medical publication database. Due to a high number of venues with few publications, we only consider the top 78 venues (the same number as for the AI dataset, making the performance metrics comparable), which account for about 10 % of the publications.

In general, we only keep publications where no input information is missing. Table 1 gives an

---

[3]Release from 2019-01-31.
[4]Code to reproduce the data will be published.

overview of both datasets. From both datasets we randomly sample 80 % as training, 10 % as validation and 10 % as test sets in a stratified manner. While our approach might favor larger conferences with this sampling strategy, we argue that this procedure better reflects the venue landscape. Larger conferences usually cover a larger thematic scope and accept more manuscripts.

5 Experimental Setup

Given the training, validation, and test splits of both datasets described in Section 4, we train all models on the training dataset and perform hyperparameter optimization on the validation data. We then report several metrics, detailed below, on the test dataset.

Hyperparameters For WTS, we perform random search (Bergstra and Bengio, 2012) with 100 runs choosing the model with the best validation Accuracy. Since our method uses a CNN that can have different filter sizes in the convolution layer, we randomly sample up to three different filter sizes from $\{1, \ldots, 9\}$. The number of filters per size are drawn from $\{5, \ldots, 300\}$. Also, we randomly choose the optimizer from $\{$Adam, SGD$\}$ (Kingma and Ba, 2014), the learning rate from continuous values in $[0.01, 0.1]$, and the dropout probability from $[0.1, 0.5]$. Other parameters such as the lengths to which the three inputs are padded and the early stopping patience on the validation loss were set manually. Table 2 shows the final parameters.

	Method	Acc	Acc@5	MRR
AI	Random	0.013	0.064	0.063
	Majority	0.086	0.319	0.212
	Log. Reg.	0.487	0.811	0.628
	Iana et al.	0.065	0.198	0.154
	WTS (Ours)	**0.503**	**0.831**	**0.645**
Medicine	Random	0.013	0.064	0.063
	Majority	0.069	0.213	0.157
	Log. Reg.	0.587	0.911	0.724
	Iana et al.	0.440	0.808	0.599
	WTS (Ours)	**0.659**	**0.948**	**0.782**

Table 3: Results of the baselines and our methods. Best values are displayed in bold.

Metrics   To compare our methods, we use three common ranking metrics: Accuracy, Accuracy@5, and Mean Reciprocal Rank (MRR). Accuracy@5 measures the fraction of test examples where the method correctly puts the venue in the top five ranks. MRR is defined as follows: Given a test set of queries Q for which the model returns a ranking of all items, $\text{MRR} = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \frac{1}{\text{rank}_i}$, where rank_i is the position at which the target item is ranked by the model. Always predicting the correct item at first position leads to a MRR of one, while bad models achieve a MRR closer to zero.

6   Results

Table 3 shows the results of all tested methods on the test sets of both datasets. On both datasets, WTS outperforms all baseline methods in all metrics. A Wilcoxon signed-rank test (Wilcoxon, 1992) at 1% confidence level shows a significant difference in the ranking of correct venues between WTS and the logistic regression baseline on both datasets. Together with the better MRR, this shows the superiority of our method to the baselines. In approximately 83% (AI) and 95% (Medicine) of all cases, the correct venue was in the top five.

Interestingly, compared to the Medicine dataset, the method by Iana et al. performs poorly on AI publications. We suspect this is due to the smaller size of the AI dataset and a higher skew in publication counts per venue (cf. Table 1).

7   Explainability

As a key part of WTS is explainability, we do not only want to recommend venues to the user but also explain why a certain venue was recommended. We

---

> **BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding**
>
> We introduce a new language representation model called BERT, which stands for Bidirectional Encoder Representations from Transformers. . . .

Figure 2: Excerpt from a publication. WTS highlights the words leading to its prediction "NAACL".

use the Integrated Gradients method introduced by Sundararajan et al. (2017) implemented in the PyTorch Captum library[5] to find the most influential words and phrases for the top five recommendations of the network. The method varies the input by linearly transitioning in 50 steps from the actual embedding inputs to matrices filled with zeros. Then, the gradients of the desired venue output with respect to the inputs are calculated for each of these steps. The gradients are averaged and multiplied by the initial input, giving positive or negative values to words and phrases that had positive or negative impact on the score of the desired venue.

Figure 2 visualizes an excerpt from WTS's output for the well-known BERT paper, which received the best paper award at NAACL 2019 (Devlin et al., 2019). It correctly ranks "NAACL" first. Integrated Gradients correctly identifies "Transformers" and "Language Understanding" as words that qualify this publication as an NLP paper.

8   Website

In order to make our system available to the public, we release WTS as a web service[6] where researchers can input their AI paper's information and receive recommendations for venues. The web service applies the trained CNN and explainability method and shows the top five predicted venues for the given paper along with a color-coded explanation (cf. Section 7) and venue-related links. The Accuracy@5 results described in Section 6 indicate that most of the times, a fitting venue is displayed to the user.

9   Conclusion

We have presented WTS, an NLP system that recommends scientific venues based on the title, abstract, and/or keywords of a publication. WTS is designed to provide an explanation as to why a certain venue was recommended, making it the

---

[5] https://captum.ai
[6] https://wheretosubmit.ml

first interpretable and open recommendation service for both, conferences and journals. We have shown that WTS provides strong recommendations on publications from the areas of AI and medicine.

Future work may regard evaluation: While each publication only was published at one specific venue, it might also be suitable for multiple other venues, implying that our current scores are just lower bounds for the actual performance. Improvements to the provided web service could be to make the list of venues sortable based on their deadline, impact, or other configurable factors.

Acknowledgements

References

Waleed Ammar, Dirk Groeneveld, Chandra Bhagavatula, Iz Beltagy, Miles Crawford, Doug Downey, Jason Dunkelberger, Ahmed Elgohary, Sergey Feldman, Vu Ha, Rodney Kinney, Sebastian Kohlmeier, Kyle Lo, Tyler Murray, Hsu-Han Ooi, Matthew E. Peters, Joanna Power, Sam Skjonsberg, Lucy Lu Wang, Chris Wilhelm, Zheng Yuan, Madeleine van Zuylen, and Oren Etzioni. 2018. Construction of the literature graph in semantic scholar. *CoRR*, abs/1805.02262.

James Bergstra and Yoshua Bengio. 2012. Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, 13(Feb):281–305.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Edanz. Edanz Journal Selector [online].

Elsevier. Elsevier Journal Finder [online].

Enago. Enago Open Access Journal Finder [online].

Andreea Iana, Steffen Jung, Philipp Naeser, Aliaksandr Birukou, Sven Hertling, and Heiko Paulheim. 2019. Building a conference recommender system based on scigraph and wikicfp. In *International Conference on Semantic Systems*, pages 117–123. Springer.

IEEE. IEEE Publication Recommender [online].

Sergey Ioffe and Christian Szegedy. 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*.

Journal Guide. Journal Guide [online].

Journal/Author Name Estimator. Journal/Author Name Estimator [online].

Ning Kang, Marius A Doornenbal, and Robert JA Schijvenaars. 2015. Elsevier journal finder: recommending journals for your paper. In *Proceedings of the 9th ACM Conference on Recommender Systems*, pages 261–264. ACM.

Kristian Kersting, Jan Peters, and Constantin Rothkopf. 2019. Was ist eine Professur fuer Kuenstliche Intelligenz? *arXiv preprint arXiv:1903.09516*.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751, Doha, Qatar. Association for Computational Linguistics.

Diederik Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *International Conference on Learning Representations*.

Manuscript Matcher. Manuscript Matcher [online].

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed Representations of Words and Phrases and their Compositionality. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 3111–3119. Curran Associates, Inc.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. Pytorch: An imperative style, high-performance deep learning library. In *Advances in neural information processing systems*, pages 8026–8037.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Stephen E Robertson. 1990. On term selection for query expansion. *Journal of documentation*, 46(4):359–364.

Martijn J Schuemie and Jan A Kors. 2008. Jane: suggesting journals, finding experts. *Bioinformatics*, 24(5):727–728.

SJFinder. SJFinder [online].

Springer. Springer Journal Suggester [online].

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. 2017. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 3319–3328. JMLR. org.

George Vrettas and Mark Sanderson. 2015. Conferences versus journals in computer science. *Journal of the Association for Information Science and Technology*, 66(12):2674–2684.

Frank Wilcoxon. 1992. Individual comparisons by ranking methods. In *Breakthroughs in statistics*, pages 196–202. Springer.

Wiley. Wiley Journal Finder [online].