An Experimental Analysis on Evaluating Patent Citations

Rabindra Nath Nandi
Hishab Singapore Pte. Ltd
rabindro.rath@gmail.com

Suman Kalyan Maity
Missouri S&T
smaity@mst.edu

Brian Uzzi
Northwestern University
uzzi@kellogg.northwestern.edu

Sourav Medya
University of Illinois Chicago
medya@uic.edu

Abstract

The patent citation count is a good indicator of patent quality. This often generates monetary value for the inventors and organizations. However, the factors that influence a patent receiving high citations over the year are still not well understood. With the patents over the past two decades, we study the problem of patent citation prediction and formulate this as a binary classification problem. We create a semantic graph of patents based on their semantic similarities, enabling the use of Graph Neural Network (GNN)-based approaches for predicting citations. Our experimental results demonstrate the effectiveness of our GNN-based methods when applied to the semantic graph, showing that they can accurately predict patent citations using only patent text. More specifically, these methods produce up to 94% recall for patents with high citations and outperform existing baselines. Furthermore, we leverage this constructed graph to gain insights and explanations for the predictions made by the GNNs.

1 Introduction & Related Work

Patents play a pivotal role in driving innovation and fostering economic growth. They provide a legal framework that allows inventors (e.g., companies, researchers) exclusive rights to their creations for a specified period, typically 20 years (Levin, 2004; Kitch, 1977; Encaoua et al., 2006). This exclusivity motivates the inventors and the businesses to invest in research and development, as they can benefit from their innovations.

Patent citations are important in the context of intellectual property (IP) and patent valuations and serve multiple important roles for patent examiners and applicants. Firstly, they aid patent examiners in assessing an invention's novelty and non-obviousness for granting patents to genuinely innovative creations. Secondly, they assist inventors by revealing the technological landscape and help them to refine claims and avoid any patent conflicts. Thirdly, patent citations play a significant role in assessing the value of patent portfolios, with more citations often signifying greater influence in specific industries. Further, researchers employ them to track tech trends and policy impact.

Several studies have analyzed patent value through the forward citations (Hall et al., 2001; Harhoff et al., 1999) and assessed economic value of patents (Sampat and Ziedonis, 2005; Hall et al., 2005). Previous research endeavors have explored broader patterns of knowledge transfer (Singh, 2003) through patent citations such as interactions between academia and industry via citations between academic papers and patents (Chen and Hicks, 2004). One of the relevant work involves prediction of patent value dependent on citation count from the text (Hsu et al., 2020) with regression. However, we differ in multiple ways: our study formulates a classification task, construct a semantic-based network, uses graph neural network (GNN)-based methods, and generates explanations.

In this paper, we perform an extensive empirical study on the power of patent text to predict citations. Our major contributions as follows.

Problem and data. We study the problem of patent citation prediction as a binary classification problem. Our study includes granted patents over last two decades and provides descriptive analyses on the meta-data of the patents in three major classes.

Method. We construct a patent semantic graph from the patent similarities and use graph neural network (GNN)-based methods for citation prediction. Our empirical evaluations show that the GNN-based methods can predict patent citations only using the patent text with high quality.

Explanation. The constructed graph combined with an explanation technique are used to get insights of the predictions of the GNNs.

Note that we have added more details for next sections in the Appendix along with background, related work, and additional experiments.

Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, pages 373–387
November 12-16, 2024 ©2024 Association for Computational Linguistics

2 Problem Definition and Data

We formulate the problem of patent citation prediction as a binary classification task where we classify the patents as *highly cited* or *low cited*. Let $\mathbb{P} = \{P_1, P_2, \cdots, P_m\}$ be the set of m patents. As the patent citations vary over years, we use the count of citations obtained by a patent after d years from the year of being granted. We denote the citation of the patent P_i after d years as C_d^i. In the experiments we use $d = 3, 5$, and 10 years and use these to generate different labels and thus they generate different datasets.

Our aim is to measure the impact of a patent by using the citations of the patent. We focus on predicting whether a particular patent will be *highly cited (positive, denoted by* 1*)* or *low cited (negative, denoted by* 0*)* at the time of its granting year by using the text-based information from the patent itself. The decision on whether a patent belongs to a particular class (positive or negative) is based on the distributions of the citations. We set a threshold based on the distribution. Let us assume the threshold is $x-$th percentile. Thus, we define patent citation class as positive based on whether the citation count is higher than the value at the top $x-$th percentile. Similarly, a patent belongs to a low cited class if the patent citation count is lower than the value at the bottom $x-$th percentile.

Definition 1 ***Citation Label:*** *We define the label function $y(C_d^i) \in \{0, 1\}$ of a patent P^i for the citations in next d years:*

$$y(C_d^i) = \begin{cases} 1, & \textit{if } C_d^i \geq C_{x,h} \\ 0, & \textit{if } C_d^i \leq C_{x,l} \end{cases}$$

where $C_{x,h}$ and $C_{x,l}$ denote the values at the top $x-$th percentile and the bottom $x-$th percentile respectively.

Other Class Labels. Though the above definition produces citation labels, one could design the labels other ways. Note that the above one produces an "easy to classify" dataset in the sense that the patent with high and low citations are well separated in the distribution. In the experiments, we explore other labeling settings. First, we define top $x-$th percentile as high, bottom $x-$th percentile as low and the rest as middle (we set $x = 10$ in the experiments). As the main goal is to identify high-quality or low-quality patents, we have divided the datasets and taken pair-wise classification in three different settings: High vs rest, high vs middle, middle vs low. Please see Sec. 4.2 for details.

CPC class	Description (short)	#Patents
A61	Medical or Veterinary Science	269364
H04	Electric Communication	379099
G06	Computing	340667

Table 1: #Patents in the individual CPC classes.

Our classification problem. We investigate the predictive power of the text in prediction of the quality of the patent, i.e., the patent citation count. To do so, we learn a prediction function f, where the features constructed from the patent text are given as input and the defined label $y(C_d^i)$ acts as the outcome variable.

Data. Our study includes the granted patents from the United States Patent and Trademark Office (USPTO)[1]. The number of patents grow exponentially over the years. We have included recent patents over the last two decades from 2000 to 2022 for our analysis. Our study focuses on citations which often depend on the area or topics of the invention, and thus, we consider on subcategories of patents. We consider patents under major (based on numbers) categories rather than all the patents. We follow the standard classification system for patents called the CPC categorization. We choose top three CPC classes in terms of the number of patents categorized in them. Table 1 shows the classes and the number of patents in each category. Descriptive analysis of the data is provided in the Appendix.

3 Methods

In patent citation prediction, there are two major challenges: (1) The texts in patents are not similar to the texts in research papers or news articles, (2) Our aim to build models that are explainable, i.e, we can find the reasoning behind their predictions.

Text-based AI Methods. Modern AI tools have recently gained popularity in patent analysis (Shomee et al., 2024). We use two methods to generate representations for the patent documents: Doc2Vec (Lau and Baldwin, 2016) and Patent Bert (Lee and Hsiang, 2020). These representations are used in combination with a multi-layered perceptron (MLP) for the classification tasks in the experiments. PatentBert fine-tunes a pre-trained BERT model with patent data and applies the model to the patent classification task.

[1] https://www.uspto.gov/

3.1 Graph-based AI Methods

Graph construction. We construct a graph from the semantic similarity between the patents where each node is a patent. Two nodes are connected if they have a high semantic similarity (∼0.6-0.8 – more details in A.4.2). We represent the patent documents with a 100-dimensional embeddings. These embeddings are generated from training a Doc2Vec model with approximately 200,000 patent texts which include their titles, abstracts, and claims. Edges in the graph are computed based on the semantic similarity between the nodes (patent embeddings computed above), specifically using the Doc2Vec features. An edge is created between nodes when their similarity surpasses a selected threshold.

Node Feature Representation. We use graph neural network (GNN)-based method to perform the patent citation prediction task. However, GNNs require initial features for the nodes. We again compute these features based on the patent text from two different embedding model: Doc2Vec (Lau and Baldwin, 2016) and PatentBert (Lee and Hsiang, 2020).

Graph Neural Networks. Graph Neural Networks (GNNs) (Kipf and Welling, 2016; Hamilton et al., 2017) have proven to be effective in making predictions on such graphs by learning relevant low-dimensional node representations through a message-passing mechanism. During message passing, each node ($u \in V$) updates its representation by aggregating information from itself and its set of neighbors $N(u)$. GNNs iteratively apply this aggregation scheme to refine the node representations, capturing the structural dependencies within the graph. The GNNs are effective for a wide range of prediction tasks over graphs such as node classification, link prediction, and graph classification. We use three types on GNNs for our study: GCN (Kipf and Welling, 2016), GraphSage (Hamilton et al., 2017), and Graph Transformer Network (GTN) (Yun et al., 2019).

4 Experiments

We use three types of patent data from three major CPC (Cooperative Patent Classification) classes: A61, H04, and G06. This results in nine separate datasets with three different periods of citation history from the year of 2000: (1) Citation history for 3 years (3-years-history): Patents published until 2019 as we count citation up to 2022; (2) Citation history for 3 years (5-years-history): published until 2017. (3) Citation history for 3 years (10-years-history): Patents published until 2011. Recent two years of data from each patent dataset (among the nine datasets above) are kept for testing.

4.1 Citation Prediction: Top vs Bottom

Labels. We have created the labels of positive and negative classes based on the citation count and the overall distribution. For the patents with high citations we choose top 10% patents based on citations (*positive class*), and correspondingly we choose bottom 10% patents for the negative class.

Results. Our objective is to demonstrate the efficacy of graph-based AI methods in the patent citation prediction. We present the results for different setting in Table 2 (recall of the positive class, i.e., patents with high citations). Please see the results for accuracy (Table 13) (accuracy) and F1 (Table 14) in the Appendix. respectively. From Table 2, we observe that all the methods can retrieve the patents with high citations accurately. This is a critical task, as high-quality patents can have a substantial impact on innovation, ultimately benefiting society. The results in Table 13 shows how the combination of textual semantics and the structure within the graph aids the models in understanding quality and thus leads to accurate predictions.

4.2 Citation Prediction: Different Labels

Labels. First, we define top $x-$th percentile as *high*, bottom $x-$th percentile as *low* and the rest as *middle* ($x = 10$). We have divided the datasets and taken pair-wise classification in three different setting: high vs rest (Table 3), high vs middle (Table 4), middle vs low (Table 5).

Results. As the labels are harder than the previous labels (Sec. 4.1), the graph-based models perform much better than just using MLP. The MLP baselines produce almost similar results as random (note that a random model would generate accuracy of .5). Our graph-based models produce good performance in terms of four measures in all the three settings, generating more than .7 in all the measures. Further, GSAGE and GTN are more sophisticated method than GCN (e.g., GSAGE have generalized aggregation function whereas GCN uses the mean as an aggregator (Hamilton et al., 2017)), and thus they produce better results than GCN.

Models	CPC Classes								
	A61			H04			G06		
	Citation Predictions @								
	3y	5y	10y	3y	5y	10y	3y	5y	10y
Doc2Vec-MLP	0.81	0.87	0.93	0.68	0.68	0.68	0.64	0.93	0.91
PatentBERT-MLP	0.76	0.87	0.91	0.68	0.68	0.68	0.68	0.89	0.86
Doc2Vec-GCN	0.83	0.86	0.92	0.76	0.76	0.76	0.70	0.94	0.92
Doc2Vec-GTN	0.75	0.82	0.90	0.67	0.86	0.87	0.55	0.87	0.90
Doc2Vec-GSAGE	0.78	0.84	0.93	0.71	0.90	0.87	0.62	0.91	0.90
PatentBERT-GCN	0.76	0.85	0.92	0.61	0.61	0.61	0.70	0.93	0.89
PatentBERT-GTN	0.77	0.85	0.94	0.83	0.88	0.83	0.58	0.86	0.87
PatentBERT-GSAGE	0.74	0.85	0.91	0.70	0.91	0.85	0.56	0.93	0.88

Table 2: **Recall of Highly Cited (positive class) Patents.** Our graph-based methods often produce the best results (blue) and recall greater than .75 indicating that they recognize more than 75% among the highly cited patents.

Model	Precision	Recall	F1-Score	Accuracy
PatentBERT-MLP	.55	.50	.52	.50
PatentBERT-GCN	.61	.58	.59	.58
PatentBERT-GTN	.70	.69	.70	.70
PatentBERT-GSAGE	.74	.73	.73	.73

Table 3: The citation prediction (best in blue) on *high* (positive) vs *rest* (negative) to show whether the models detect the high quality patents from the rest.

Model	Precision	Recall	F1-Score	Accuracy
PatentBERT-MLP	.55	.51	.52	.51
PatentBERT-GCN	.61	.59	.60	.69
PatentBERT-GTN	.73	.73	.73	.73
PatentBERT-GSAGE	.74	.74	.74	.74

Table 4: Citation prediction (best in blue) on *high* (positive) vs *middle* (negative) to differentiate the high quality patents from the "mediocore" ones.

4.3 Explanations with GNNs

One primary motivation for designing graph-based methods is the capability to provide explanations for the predictions (Kakkad et al., 2023; Kosan et al., 2023). Note that it is difficult to explain patent quality from the text itself with traditional methods such as LIME (Ribeiro et al., 2016) as the patent text is domain-specific and often written by an expert lawyer with a lot of jargon. Thus, our graph construction method becomes useful for generating explanations. We choose a set of 50 nodes from both the classes. GNNExplainer (Ying et al., 2019) is designed to explain the prediction behavior of GNNs while producing a subgraph as an explanation for node classification tasks. In this context, we can gain insights into the relationships between different nodes (patents) that impact citations. We compare these two sets of explanation subgraphs obtained for the nodes in both classes. We compute three graph-specific properties: density, degree, and clustering coefficient (CC). We report the average of the values from the subgraphs

Model	Precision	Recall	F1-Score	Accuracy
PatentBERT-MLP	.49	.49	.49	.51
PatentBERT-GCN	.56	.56	.56	.54
PatentBERT-GTN	.72	.71	.71	.69
PatentBERT-GSAGE	.72	.71	.71	.70

Table 5: Citation prediction (best in blue) on *middle* (positive) vs *low* (negative) to differentiate the "mediocore" patents from the low-quality ones.

in both classes. Table 6 shows the results. Clearly, average clustering coefficient can distinguish between the explanation subgraphs of highly cited patents from the explanation subgraphs of the low cited ones. This indicates that the neighborhood of the highly cited patents are densely connected.

Data	Label	Citations	Avg. Density	Avg. Degree	CC
A61	1	high	0.470	5.705	0.265
A61	0	low	0.563	6.232	0.228
H04	1	high	0.322	16.22	0.46
H04	0	low	0.287	10.826	0.331
G06	1	high	0.221	14.368	0.431
G06	0	low	0.221	9.21	0.284

Table 6: Comparison of graph-based properties in the explanation subgraphs for nodes in both classes. CC denotes average clustering co-efficient of the nodes.

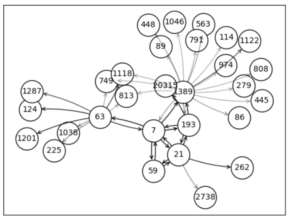


Figure 1: Explanation subgraph of node 7 with the patent titled *"Interchangeable shaft assemblies for use with a surgical instrument"* produced by the GNN-explainer method (Ying et al., 2019). Please refer to Table 7 for the specific patent-related information.

4.3.1 Example Explanation Subgraph

We show an example of the explanation subgraph that is obtained from our framework with the GN-

Node ID	Connection	Title	Citations
7	Self	Interchangeable shaft assemblies for use with a surgical instrument	508
21	Direct	Modular powered surgical instrument with detachable shaft assemblies	592
59	Direct	Drive system lockout arrangements for modular surgical instruments	538
63	Direct	Rotary powered articulation joints for surgical instruments	531
193	Direct	Locking arrangements for detachable shaft assemblies	409
1389	Direct	Robotically powered surgical device with manually-actuatable reversing system	117
1122	Indirect	Shaft assembly arrangements for surgical instruments	153
20315	Indirect	Articulation mechanism for surgical instrument	1206
1287	Indirect	Surgical device having multiple drivers	195
1201	Indirect	Hand held rotary powered surgical instruments with end effectors	142
1118	Indirect	Articulatable surgical instrument configured for detachable use with a robotic system	153

Table 7: Information on patents/nodes of example explanation subgraph in Fig. 1. We observe that explanation subgraph attached to a highly cited node/patent consists of nodes/patents that are highly cited. Interestingly, all the nodes that are both indirectly or directly connected to the node/patent being explained have high citations.

Model	2000-2004				2005-2009				2010-2014			
	Acc	Pr	Re	F1	Acc	Pr	Re	F1	Acc	Pr	Re	F1
Doc2vec-GCN	0.61	0.66	0.69	0.67	0.66	0.67	0.84	0.74	0.68	0.68	0.89	0.77
Doc2Vec-GTN	0.65	0.72	0.66	0.69	0.67	0.68	0.83	0.75	0.69	0.68	0.89	0.77
Doc2Vec-GSAGE	0.65	0.72	0.67	0.70	0.66	0.0.68	0.81	0.74	0.71	0.70	0.88	0.78
PatentBert-GCN	0.66	0.73	0.67	0.70	0.69	0.73	0.75	0.74	0.72	0.73	0.85	0.78
PatentBert-GTN	0.67	0.76	0.65	0.70	0.70	0.74	0.76	0.75	0.73	0.75	0.82	0.78
PatentBert-GSAGE	0.67	0.76	0.64	0.69	0.69	0.74	0.73	0.73	0.73	0.75	0.83	0.78

Table 8: Results on the test dataset with patents only from the year of 2016 in A61 where *Acc* denotes accuracy and *Pr, Re, F1* denote Precision, Recall and F1-score for the positive class. We construct three different training sets from a span of 5-years from 2000-2014. The results show that training with recent patents have a more accurate prediction of citation classes for the future patents.

NExplainer method (Ying et al., 2019). In Figure 1, we present the subgraph resulting from the explanation of the patent titled *"Interchangeable shaft assemblies for use with a surgical instrument"* (node with the index 7). Note that there are several nodes that are directly connected (with the dark edges). The graph edges are color-coded to convey their strength: black edges represent strong connections, while the shadow lines indicate weaker connections. We extract the critical subgraph nodes based on the presence of black edge lines, signifying their importance in the explanation subgraph.

Furthermore, to understand the example of the patents in the explained subgraph, we present the patent title, the number of citations, and the connection type in Table 7. The focal patent (node 7) is highly cited patent with 508 citations. Notably, both directly and indirectly connected nodes also have titles related to surgical devices and instruments same as the focal node, with high citation counts. This explainer subgraph example suggests that the number of citations in the similar patents might indirectly impact the number of citation of the focal patent, even though our proposed GNNs do not use this information for the prediction.

4.4 Impact of *Recency* on Citations

We demonstrate that the recency of the patents are useful for patent citation prediction. Here we evaluate the influence of patents from recent years within the A61 CPC class. We utilize three distinct training sets with five years of patents: 2000-2004, 2005-2009, and 2010-2014, respectively. The test set remain consistent across all experiments with patents from 2016. The results, presented in Table 8, indicate that training with more recent patents enhances the models' predictive capabilities of citation classes for the future patents. For instance, when using the PatentBert-GSAGE approach, we achieve higher levels of accuracy, precision, recall, and F1-score when training with patents from 2010-2014 to predict citations for patents in 2016.

5 Discussions

We draw several key takes from the study. *(1) Text and network structure matter:* Graph-based AI models (GNNs) can predict patent citation accurately only from the text of title, abstract, and claims. Understanding the network structure of the patent landscape is also important. *(2) Explanation is the key:* Though several deep learning models have good predictive power, they might lack domain-specific explanations and the GNN-based explainers might be helpful. *(3) Recent data is important:* The text from recent patents are more useful for citation prediction, thus, models should be mindful about the training data and possibly need re-training regularly.
Code and data are accessible at https://github.com/robi56/patent_high_citation/.

6 Ethical considerations

In this work, we have built AI models based on textual information and patent semantic network to predict patent citations after the patents are granted. We do not foresee any ethical issues from our study.

7 Limitations

This paper addresses a timely subject related to AI-based methods to predict patent citations. The dataset and the model used for this study are publicly available. While the paper shows the capability graph-based approaches towards patent citation prediction, one could further investigate the reasoning on patents getting high citations and build a few prototypes.

References

Sophia Althammer, Mark Buckley, Sebastian Hofstätter, and Allan Hanbury. 2021. Linguistically informed masking for representation learning in the patent domain. *arXiv preprint arXiv:2106.05768*.

Juho Bai, Inwook Shim, and Seog Park. 2020. Mexn: Multi-stage extraction network for patent document classification. *Applied Sciences*, 10(18):6229.

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

Fernando Benites, Shervin Malmasi, and Marcos Zampieri. 2018. Classifying patent applications with ensemble methods. *arXiv preprint arXiv:1811.04695*.

Gaetano Cascini and Manuel Zini. 2008. Measuring patent similarity by comparing inventions functional trees. In *Computer-Aided Innovation (CAI) IFIP 20th World Computer Congress, Proceedings of the Second Topical Session on Computer-Aided Innovation, WG 5.4/TC 5 Computer-Aided Innovation, September 7-10, 2008, Milano, Italy*, pages 31–42. Springer.

Alok K Chakrabarti, Israel Dror, and Nopphdot Eakabuse. 1993. Interorganizational transfer of knowledge: an analysis of patent citations of a defense firm. *IEEE Transactions on Engineering Management*, 40(1):91–94.

Chaomei Chen and Diana Hicks. 2004. Tracing knowledge diffusion. *Scientometrics*, 59(2):199–211.

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*.

David Encaoua, Dominique Guellec, and Catalina Martínez. 2006. Patent systems for encouraging innovation: Lessons from economic analysis. *Research policy*, 35(9):1423–1440.

Lintao Fang, Le Zhang, Han Wu, Tong Xu, Ding Zhou, and Enhong Chen. 2021. Patent2vec: Multi-view representation learning on patent-graphs for patent classification. *World Wide Web*, 24(5):1791–1812.

Sijie Feng. 2020. The proximity of ideas: An analysis of patent text using machine learning. *PloS one*, 15(7):e0234880.

Mattyws F Grawe, Claudia A Martins, and Andreia G Bonfante. 2017. Automated patent classification using word embedding. In *2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 408–411. IEEE.

Bronwyn H. Hall, Adam Jaffe, and Manuel Trajtenberg. 2005. Market value and patent citations. *The RAND Journal of Economics*, 36(1):16–38.

Bronwyn H Hall, Adam B Jaffe, and Manuel Trajtenberg. 2001. The nber patent citation data file: Lessons, insights and methodological tools. Working Paper 8498, National Bureau of Economic Research.

William L. Hamilton, Rex Ying, and Jure Leskovec. 2017. Inductive representation learning on large graphs. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 1025–1035. Curran Associates Inc.

Dietmar Harhoff, Francis Narin, F. M. Scherer, and Katrin Vopel. 1999. Citation Frequency and the Value of Patented Inventions. *The Review of Economics and Statistics*, 81(3):511–515.

Po-Hsuan Hsu, Dokyun Lee, Prasanna Tambe, and David H. Hsu. 2020. Deep learning, text, and patent valuation.

Junegak Joung and Kwangsoo Kim. 2017. Monitoring emerging technologies for technology planning using technical keyword based analysis from patent data. *Technological Forecasting and Social Change*, 114:281–292.

Jaykumar Kakkad, Jaspal Jannu, Kartik Sharma, Charu Aggarwal, and Sourav Medya. 2023. A survey on explainability of graph neural networks. *arXiv preprint arXiv:2306.01958*.

MMS Karki. 1997. Patent citation analysis: A policy analysis tool. *World Patent Information*, 19(4):269–272.

Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.

Edmund W Kitch. 1977. The nature and function of the patent system. *the Journal of Law and Economics*, 20(2):265–290.

Mert Kosan, Samidha Verma, Burouj Armgaan, Khushbu Pahwa, Ambuj Singh, Sourav Medya, and Sayan Ranu. 2023. Gnnx-bench: Unravelling the utility of perturbation-based gnn explainers through in-depth benchmarking. *arXiv preprint arXiv:2310.01794*.

Jey Han Lau and Timothy Baldwin. 2016. An empirical evaluation of doc2vec with practical insights into document embedding generation. *arXiv preprint arXiv:1607.05368*.

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. 2019. Patentbert: Patent classification with fine-tuning a pre-trained bert model. *arXiv preprint arXiv:1906.02124*.

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

Richard Levin. 2004. A patent system for the 21st century. *Issues in Science and Technology*, 20(4):49–54.

Shaobo Li, Jie Hu, Yuxin Cui, and Jianjun Hu. 2018. Deeppatent: patent classification with convolutional neural networks and word embedding. *Scientometrics*, 117:721–744.

Rui Meng, Sanqiang Zhao, Shuguang Han, Daqing He, Peter Brusilovsky, and Yu Chi. 2017. Deep keyphrase generation. *arXiv preprint arXiv:1704.06879*.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Shervin Minaee, Nal Kalchbrenner, Erik Cambria, Narjes Nikzad, Meysam Chenaghlu, and Jianfeng Gao. 2021. Deep learning–based text classification: a comprehensive review. *ACM computing surveys (CSUR)*, 54(3):1–40.

Diego Mollá and Dilesha Seneviratne. 2018. Overview of the 2018 alta shared task: Classifying patent applications. In *Proceedings of the Australasian Language Technology Association Workshop 2018*, pages 84–88.

Francis Narin. 1994. Patent bibliometrics. *Scientometrics*, 30(1):147–155.

Heeyong Noh, Yeongran Jo, and Sungjoo Lee. 2015. Keyword selection and processing strategy for applying text mining to patent analysis. *Expert Systems with Applications*, 42(9):4348–4360.

Charles Oppenheim. 2000. Do patent citations count. *The web of knowledge: A festschrift in honor of Eugene Garfield*, pages 405–432.

Subhash Chandra Pujari, Annemarie Friedrich, and Jannik Strötgen. 2021. A multi-task approach to neural multi-label hierarchical patent classification using transformers. In *Advances in Information Retrieval: 43rd European Conference on IR Research, ECIR 2021, Virtual Event, March 28–April 1, 2021, Proceedings, Part I 43*, pages 513–528. Springer.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144.

Julian Risch and Ralf Krestel. 2019. Domain-specific word embeddings for patent classification. *Data Technologies and Applications*, 53(1):108–122.

Bhaven N. Sampat and Arvids A. Ziedonis. 2005. *Patent Citations and the Economic Value of Patents*, pages 277–298. Springer Netherlands, Dordrecht.

Homaira Huda Shomee, Zhu Wang, Sathya N Ravi, and Sourav Medya. 2024. A comprehensive survey on ai-based methods for patents. *arXiv preprint arXiv:2404.08668*.

Jasjit Singh. 2003. Social networks as drivers of knowledge diffusion. Technical report, Technical report.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph attention networks. In *International Conference on Learning Representations*.

Arnold Verbeek, Koenraad Debackere, and Marc Luwel. 2003. Science cited in patents: A geographic" flow" analysis of bibliographic citation patterns in patents. *Scientometrics*, 58(2):241–263.

Zhitao Ying, Dylan Bourgeois, Jiaxuan You, Marinka Zitnik, and Jure Leskovec. 2019. Gnnexplainer: Generating explanations for graph neural networks. *Advances in neural information processing systems*, 32.

Yongmin Yoo, Cheonkam Jeong, Sanguk Gim, Junwon Lee, Zachary Schimke, and Deaho Seo. 2023. A novel patent similarity measurement methodology: Semantic distance and technological distance. *arXiv preprint arXiv:2303.16767*.

Kenneth A Younge and Jeffrey M Kuhn. 2016. Patent-to-patent similarity: A vector space model. *Available at SSRN 2709238*.

Seongjun Yun, Minbyul Jeong, Raehyun Kim, Jaewoo Kang, and Hyunwoo J Kim. 2019. Graph transformer networks. *Advances in neural information processing systems*, 32.

Manzil Zaheer, Guru Guruganesh, Kumar Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, et al. 2020. Big bird: Transformers for longer sequences. *Advances in neural information processing systems*, 33:17283–17297.

A Appendix

A.1 Background

Types of Patents. In the United States, there are *three major types* of patents granted by the United States Patent and Trademark Office (USPTO). These patents are designed to protect different kinds of inventions and intellectual property. **(1) Utility Patents:** Utility patents are the most common type of patent and cover new and useful processes, machines, manufactured articles, and compositions of matter.**(2) Design Patents:** Design patents protect the ornamental or aesthetic design of a functional item. They are often sought for products with unique visual characteristics, such as consumer electronics, jewelry, and automotive parts. **(3) Plant Patents:** Plant patents protect new and distinct varieties of plants that have been asexually reproduced (e.g., through cuttings or grafting).

Components of Utility Patents. In this work we mainly focus on utility patents. Utility patents, also known as "patents for inventions", protect new and useful processes, manufactured articles, and compositions of matter. We use these key components of a utility patent in our study: (1) Title: The title provides a concise and descriptive name for the invention. (2) Abstract: An abstract is a concise summary of the invention, typically limited to 150-250 words. It provides a brief overview of the invention's technical aspects and applications. (3) Claims: The claims define the legal boundaries of the patent. They precisely describe the elements or steps that make the invention unique and patentable. We use the text of title, abstract and claims to create features for our patent citation prediction task. The claims have been to useful for other task such as CPC (topic-based) classification (Lee and Hsiang, 2020). Title and abstracts are often used in similar natural language processing tasks such as keyphrase generation (Meng et al., 2017).

Importance of Patent Citations. Patent citations, which refer to the references to prior patents within a newly granted patent, serve several purposes and are important for various stakeholders in the intellectual property ecosystem. (1) Assessment of Novelty and Non-obviousness: Patent examiners use patent citations to assess the novelty and non-obviousness of a new invention. By examining the references cited in a patent application, examiners can determine whether the claimed invention is truly novel and represents a non-obvious advancement over prior art. This is a fundamental step in the patent examination process and helps ensure that only truly innovative inventions receive patent protection. (2) Prior Art Search: For inventors and patent applicants, reviewing patent citations can aid in understanding the existing landscape of related technologies and inventions, often referred to as "prior art." This can help inventors refine their claims, identify gaps in existing knowledge, and potentially avoid pursuing inventions that are unlikely to be granted patents due to the existence of prior art. (3) Patent Valuation: A patent with numerous citations from other patents may be considered more valuable because it indicates that the patented technology is widely recognized as influential or relevant within a specific industry or field.

In summary, patent citations are essential for the evaluation and utilization of intellectual property. They provide valuable information about the state of innovation, the relationship between patents, and the technological advancements within specific fields. Thus, we focus on building AI-based models to predict the citations.

A.2 Related Work

Patent classification. Recent advancements in Machine Learning have led to the application of various ML techniques aimed at enhancing the efficiency of patent classification. Benites et al. (Benites et al., 2018) presented a top-performing solution in the ALTA 2018 Shared Task on patent classification (Mollá and Seneviratne, 2018), utilizing the full text of patent documents. Grawe et al. (Grawe et al., 2017) employed an LSTM in conjunction with word embeddings for classification. Risch and Krestel (Risch and Krestel, 2019) pre-trained fastText word embeddings using a substantial corpus of patent documents, integrating them with Gated Recurrent Units (GRUs) for classification. Li et al. (Li et al., 2018) proposed DeepPatent, which is a deep learning algorithm based on convolutional neural networks. PatentBERT (Lee and Hsiang, 2019) focuses on fine-tuning a pre-trained BERT (Devlin et al., 2018) model which uses only the first claim of a patent and achieving noteworthy results. Patent2vec (Fang et al., 2021) adopted a multi-view graph-based approach with tags to patent classification. Bai et al. (Bai et al., 2020) proposed a Multi-stage Feature Extraction Network (MEXN), comprising a paragraph encoder and summarizer for all patent paragraphs to enhance classification. Pujari et al. (Pujari et al., 2021) developed a hierarchical transformer-based multi-task model that trained an intermediate SciBERT (Beltagy et al., 2019) layer using title and abstract as input text. In a comparative analysis of BERT and SciBERT for patent classification, Althammer et al. (Althammer et al., 2021) discovered that the SciBERT model outperformed BERT. Zaheer et al. propose Big Bird (Zaheer et al., 2020), a long-text transformer, and apply it to patent classification by incorporating title, abstract, and claims into the classification process.

Patent Similarity. Measuring similarity between patents has become another prominent field of research involving patents. Consequently, a substantial body of research has concentrated on methodological aspects, employing machine learning and deep learning, particularly natural language processing (NLP) techniques, to gauge patent similarity. Cascini and Zini (Cascini and Zini, 2008) introduced a clustering algorithm that evaluates patent similarity by taking into account hierarchical and functional interactions among patents. Vector space models have also been utilized in patent analysis. Younge et al. (Younge and Kuhn, 2016) developed a single vector space-based model for automatically measuring the continuous similarity distance between pairs of patents. Feng (Feng, 2020) devised a similarity measurement technique using vector space representations of patent abstracts with Document Vectors (Doc2Vec) (Le and Mikolov, 2014). Noh and Lee applied text mining to patent analysis by employing keyword selection and processing strategies (Noh et al., 2015). Similarly, Joung and Kim adopted a keyword-based approach for technology planning (Joung and Kim, 2017). Recently, Yoo et al. (Yoo et al., 2023) proposed a hybrid method that automatically assesses patent similarity, taking into account both semantic and technological similarities.

Patent Citations. Patent citations serve as a significant metric to gauge intellectual heritage and influence. They have been employed to assess the dissemination and exchange of knowledge in research and development, as well as to measure research productivity and impact (Narin, 1994). The information derived from patent citations can effectively portray the transmission of knowledge (Karki, 1997; Oppenheim, 2000). Previous investigations have delved into the broader patterns of knowledge transfer through patent citations. For instance, Chakrabarti et al. (Chakrabarti et al., 1993) scrutinized inter-organization patent citation trends in defense-related research and development transitioning into the civilian sector. Chen and Hicks (Chen and Hicks, 2004) examined the interactions between academia and industry by scrutinizing citations between academic papers and patents in the field of tissue engineering. Verbeek et al. (Verbeek et al., 2003) explored the geographic distribution of scientific research's influence on patents in the domains of biotechnology and information technology. Singh (Singh, 2003) investigated how the social distance between inventors impacts the flow of knowledge within USPTO patents. These studies on knowledge diffusion were primarily based on the citation patterns between pairs of entities.

A.3 Data

Our study includes the granted (accepted) patents from the United States Patent and Trademark Office (USPTO)[2]. The number of patents grow exponentially over the years. We have included recent patents over the last two decades from 2000 to 2022 for our analysis. Our study focuses on citations which often depend on the area or topics of the invention. This fact naturally leads us to focus on subcategories of patents. For a better understanding on how patents are cited as well to build better models to predict the citations, we consider patents under major (based on numbers) categories rather than all the patents. We follow the standard classification system for patents called the CPC categorization[3]. We choose top three CPC classes in terms of the number of patents categorized in them. Table 1 shows the classes and the number of patents in each category.

We show descriptive analysis of the data on the distribution of several important components of patents for the three CPC classes. Fig. 2 shows statistics for all the three major CPC classes (A61, H04, and G06) on average number of inventors (team size), figures, sheets. One interesting observation is that A61 (i.e., patents in the medical domain) has higher average then the other two for all the years. Over the years, all the values have an upward trend. Upward trend in team size implies the collaboration is increasing over the years. On

[2] https://www.uspto.gov/

[3] https://www.uspto.gov/web/patents/classification/cpc/html/cpc.html

the other hand, Fig. 3 shows statistics on the average number of claims (all and dependent) for all the three major CPC classes. Note that the claims describe the elements or steps that make the invention unique and patentable. Interestingly, in all areas, the number of claims look similar. Over the years, all the values have mostly a downward trend indicating that number of claims might not be a driving factor to get a patent accepted.

A.4 Methods

There are numerous deep learning-based methods that have been proposed for text classification tasks (Minaee et al., 2021; Shomee et al., 2024). However, in the patent citation prediction tasks, there are two major challenges:

- The texts in patents are not similar to the texts in research papers or news articles.
- Our aim to build models that are explainable, i.e, we can find the reasoning behind their predictions. Furthermore, we aim to understand the mechanism behind a patent getting high citations.

A.4.1 Text-based AI Methods

We use two methods to generate representations for the patent documents from traditional text-based AI or NLP models: Doc2Vec and Patent Bert. Note that these representations are used in combination with a multi-layered perceptron (MLP) for the classification tasks in the experiments. We describe these two methods one being generic and another focusing on patent data:

- Doc2Vec (Le and Mikolov, 2014): Doc2Vec, also known as Paragraph Vector, is an extension of Word2Vec (Mikolov et al., 2013), a popular method (Lau and Baldwin, 2016) for representing paragraphs in stead of words as a vector representation in natural language processing (NLP). While Word2Vec learns vector representations for words, Doc2Vec goes a step further by learning representations for entire documents or paragraphs while capturing the semantic meaning and context of a document. Each document is represented as a fixed-length vector. We use the representations produced by Doc2Vec and feed them through an MLP to predict the citation class of a patent.
- PatentBert (Lee and Hsiang, 2020): This method fine-tunes a pre-trained BERT model and applies it to the task of patent classification. It uses the BERT based pre-trained model for fine-tuning.

A.4.2 Graph-based AI Methods

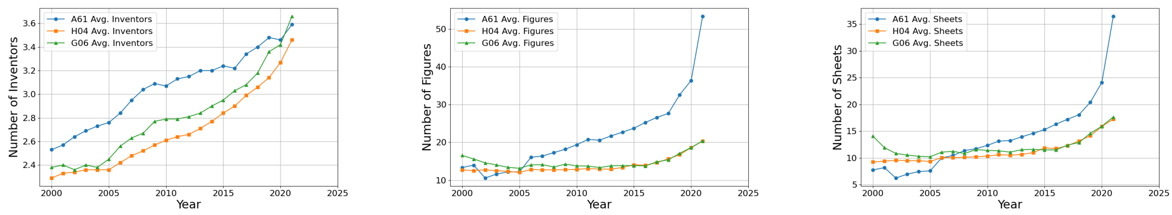
Graph construction. We construct a graph from the semantic similarity between the patents where each node is a patent. Two nodes are connected if they have a high semantic similarity.

Proximity creation via training the Doc2Vec model: We represent the patent documents with a 100-dimensional vector representations (embeddings). These embeddings are generated from training a Doc2Vec model with approximately 200,000 patent texts which include their titles, abstracts, and claims. These embeddings are designed to capture the semantic similarity between patent text data, thus will help us to create the edges between patents.

Edge Construction: Edges in the graph are computed based on the semantic similarity between the nodes (patent embeddings computed above), specifically using the Doc2Vec features. An edge is created between nodes when their similarity surpasses a selected threshold, typically falling within the range of 0.62 to 0.8. The choice of the similarity threshold is based on the desired density of the graph, which we vary from 5 to 25.

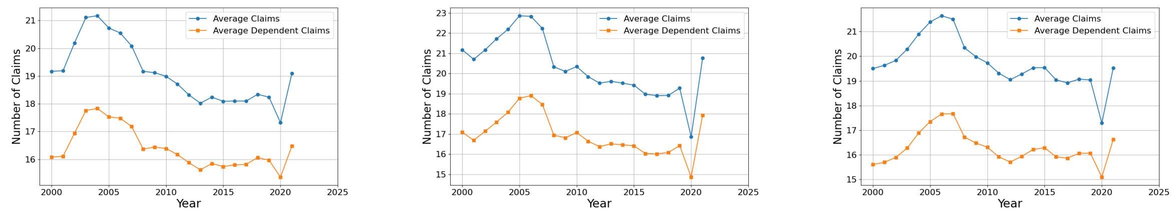
Node Feature Representation: We use graph neural network (GNN)-based method to perform the patent citation prediction task. However, GNNs require initial features for the nodes. We again compute these features based on the patent text. Specifically, we create two distinct types of node features from two different embedding model (we use these features separately in the experiments):

- **Features from Doc2Vec:** The first type of node features is generated using the Doc2Vec model trained in the previous step. These features are calculated based on the semantic content of the patent text data.
- **Features from PatentBert (Lee and Hsiang, 2020):** The second type is obtained from the PatentBert model which is trained on a dataset comprising over 100 million patents, including international patents. This model, based on BERTLARGE (Devlin et al., 2018), produces 1024-dimensional feature representations.



(a) Average #Inventors (b) Average #Figures (c) Average #Sheets

Figure 2: Descriptive statistics for all the three major CPC classes (A61, H04, and G06) on (a) Average number of inventors (team size), (b) Average number of figures, and (c) Average number of Sheets. Interestingly, A61 has higher average then the other two for all the years. Over the years, all the values have an upward trend.



(a) A61 (b) G06 (c) H04

Figure 3: Descriptive statistics on average number of claims (all and dependent) for all the three major CPC classes (a) A61, (b) H04, and (c) G06. Note that the claims describe the elements or steps that make the invention unique and patentable. Interestingly, in all areas, the number of claims look similar. Over the years, all the values have mostly a downward trend.

Graph Neural Networks. Consider a graph, denoted as $G = (V, X, A)$, consisting of a set of nodes (V) and a set of edges (E). Let $X \in \mathbb{R}^{n \times d}$ represent the d-dimensional features of n nodes in V, while $A \in \{0, 1\}^{n \times n}$ is the adjacency matrix specifying edges in the edge set E. Graph Neural Networks (GNNs) (Kipf and Welling, 2016; Hamilton et al., 2017; Veličković et al., 2018) have proven to be effective in making predictions on such graphs by learning relevant low-dimensional node representations through a message-passing mechanism.

During message passing, each node ($u \in V$) updates its representation by aggregating information from itself and its set of neighbors $N(u)$. Mathematically, the update in l-th step can be represented as follows:

$$h_u^{(l)} = AGGR(h_u^{(l-1)}, \{h_i^{(l-1)} | i \in N(u)\}) \quad (1)$$

where $h_u^{(l)}$ is the updated representation of node u at iteration l, obtained by applying the aggregation operation ($AGGR$) to combine its previous representation ($h_u^{(l-1)}$) with those of its neighboring nodes. The representation at the 0-th step is the initial feature set of the nodes. GNNs iteratively apply this aggregation scheme to refine the node representations, capturing the structural dependencies within the graph. The GNNs are effective for a wide range of prediction tasks over graphs such as node classification, link prediction, and graph classification. We use three types on GNNs for our study.

(1) GCN (Kipf and Welling, 2016): In the message passing framework, GCN uses sum as its Aggregation function. The propagation rule is as follows:

$$H^{(l)} = \sigma(D^{-1/2}\tilde{A}D^{-1/2}H^{(l-1)}W^{(l-1)}) \quad (2)$$

where $\tilde{A} = A + I$ is the adjacency matrix with self connections. $W^{(l-1)}$ is layer specific weight matrix. σ is the activation function. H^l is matrix of activation in l th layer. In theory, GCN considers spectral convolution on graph as a multiplication of signal and filter.

(2) GraphSage (Hamilton et al., 2017): GraphSage extends the ideas of message aggregation in two important ways. First, considers multiple aggregator functions like mean, element wise max pooling and LSTM. Second, it concatenates node's current representation with the aggregated neighborhood vector.

$$AGG_u^{l-1} = AGG(h_i^{(l-1)} | i \in N(u)\ ))$$
$$h_u^{(l)} = \sigma(WConcat(h_u^{(l-1)}, AGG_u^{l-1})$$

(3) Graph Transformer Network (GTN) (Yun et al., 2019): Graph Transformer Networks (GTN) uses self-attention mechanisms to capture relationships between the nodes in the graph. This self-attention mechanism makes it more effective in the traditional prediction tasks over graphs. When X is the set of node features, we can represent the node embeddings as $H = \text{Enc}(X)$, where Enc is the encoding function, typically based on self-attention mechanisms. The self-attention mechanism computes attention scores between nodes and combines their features accordingly:

$$\text{Att}(H) = \sigma\left(\frac{(H \cdot W_q)(H \cdot W_k)^T}{\sqrt{d_k}}\right) \cdot (H \cdot W_v)$$

Here, Att, σ, W_q, W_k, and W_v are Attention, softmax function, learnable weight matrices, and d_k is the dimension of the key vectors. GTNs often employ multi-head attention, which allows the model to focus on different aspects of the graph simultaneously. The final output from the self-attention mechanism is typically used to perform a graph convolution operation. This operation aggregates information from neighboring nodes to update node features. The graph convolution can be represented as: $\text{GraphConv}(H) = \sigma\left(\text{MultiHead}(H) \cdot W_o\right)$. Here, σ is the activation function, and W_o is another learnable weight matrix.

A.5 Experimental Settings

A.5.1 Dataset

The focus of the study is on building methods to predict a quality of a patent from its citations. Essentially, we aim to classify patents with high and low citations. Thus, given a new patent we would be predict whether the patent will have high or low citations. We use three types of patent data that are prepared for three major CPC (Cooperative Patent Classification)[4] classes: A61, H04, and G06. This results in nine separate datasets with three different periods of citation history from the year of 2000: (1) Citation history for 3 years (3-years-history): Patents published until 2019 as we can count citation up to 2022; (2) Citation history for 3 years (5-years-history): Patents published until 2017. (3) Citation history for 3 years (10-years-history): Patents published until 2011.

[4]https://www.uspto.gov/web/patents/classification/cpc/html/cpc.html

Dataset Preparation for Classification Task Recent two years of data from each patent dataset (among the nine datasets above) are kept for testing. The remaining data is used for model training. This test dataset is designed to understand the model behavior to predict citation behavior on new unseen patents.

A.5.2 Training and Test Data Split

We consider several variations in splitting the Training and Test dataset corresponding various experiments. The detailed description on class sizes and train/test splits are provided in Tables 9,10,11,12. We have considered different cut-off thresholds to determine a patent to be highly cited or lowly cited. The cut-offs corresponding to experiments in Table 2 for highly cited patents in the A61, H04, and G06 CPC classes are 18, 15, and 16 citations respectively, with patents below these cut-offs are considered lowly cited.

Table 9: Train/Test Data Distribution corresponding to experiments in Table 2: 3-years-history (Train: 2000-2017, Test: 2018-2019), 5-years-history (Train: 2000-2015, Test: 2016-2017), 10-years-history (Train: 2000-2010, Test: 2011-2012)

CPC Classes	Years	(Train, Test)
A61	3y	(40874, 2624)
A61	5y	(37799, 3142)
A61	10y	(20912, 5981)
H04	3y	(59713, 1986)
H04	5y	(51683, 7306)
H04	10y	(25699, 11023)
G06	3y	(52701, 2389)
G06	5y	(46521, 5123)
G06	10y	(21707, 8925)

A.5.3 Performance Measures

We compare the proposed methods with three following performance measures: accuracy, recall of positive class (high citations), and F1-Score on positive class.

- **Accuracy:** It measures how well the models performs in correctly classifying all patents including both with high citations and low citations.
- **Recall of positive class (high citations):** One of our major goals is to retrieve the high qual-

Table 10: Class Size Distribution for A61 CPC Classes corresponding to experiments in Tables 3, 4, 5 (Train: 2000-2015, Test: 2016)

Category	Train (high, low)	Test (high, low)	Total (Train, Test)
(Top, Middle)	(8996, 8996)	(2157, 1078)	(17992, 3235)
(Top, Bottom)	(9004, 8996)	(2148, 1078)	(18000, 3226)
(Middle, Bottom)	(8996, 8996)	(2157, 2157)	(17992, 4314)

Table 11: Train/Test data distribution corresponding to experiments in Table 8.

Period	Train	Test
2000-2004	10933	2687
2005-2009	8132	2687
2010-2014	13198	2687

Table 12: Yearly Distribution of Patent Selection for experiments in Table 2.

Year	A61	H04	G06
2000	188	655	489
2001	944	2804	2118
2002	687	2905	2144
2003	713	2984	2277
2004	1487	3468	2458
2005	2978	3017	2368
2006	3677	4604	3552
2007	3132	3884	3120
2008	2826	3770	3356
2009	3039	3971	3681
2010	4494	4758	5045
2011	4309	4650	4837
2012	4946	5330	5568
2013	5007	5321	5818
2014	4543	4830	4973
2015	2903	2819	3016
2016	1915	1682	1811
2017	1312	7594	4659

ity patents. Thus, we use *Recall* for the patents with high citations. It measures the model's ability to identify patents with high citation out of all patents with high citations. A high recall would suggest that the model has high capability to identify high-quality patents.

- **F1-Score on positive class:** This assesses the model's ability to accurately predict patents with high citations while balancing between precision and recall: $F1_{\text{positive}} = \frac{2 \cdot \text{Precision}_{\text{positive}} \cdot \text{Recall}_{\text{positive}}}{\text{Precision}_{\text{positive}} + \text{Recall}_{\text{positive}}}$.

A.5.4 Other Settings

All experimental work has been conducted with a Google Cloud Ubuntu virtual machine with 64 GB of RAM and 8 vCPUs (equivalent to 4 physical CPU cores). We have also set the maximum number of epochs to 500, the optimizer as Adam optimizer, weight decay of $5e^{-4}$, loss function as the cross-entropy function. We systematically vary the learning rate across a range from 0.01 to 0.00001 to explore how different learning rates affect the model's convergence and performance.

A.6 Reproducibility and Code

We have developed a publicly accessible codebase (https://github.com/robi56/patent_high_citation/). We believe that it will help practitioners either implement the techniques in practice or use them as competing baselines.

Models	CPC Classes								
	A61			H04			G06		
	Citation Predictions @								
	3y	5y	10y	3y	5y	10y	3y	5y	10y
Doc2Vec-MLP	0.77	0.85	0.75	0.68	0.68	0.68	0.64	0.63	0.57
PatentBERT-MLP	0.74	0.85	0.89	0.69	0.69	0.69	0.68	0.66	0.67
Doc2Vec-GCN	0.78	0.84	0.75	0.73	0.73	0.73	0.69	0.62	0.57
Doc2Vec-GTN	0.74	0.81	0.75	0.69	0.61	0.57	0.58	0.66	0.60
Doc2Vec-GSAGE	0.76	0.82	0.76	0.71	0.57	0.59	0.64	0.65	0.61
PatentBERT-GCN	0.74	0.83	0.75	0.64	0.64	0.64	0.68	0.64	0.63
PatentBERT-GTN	0.75	0.84	0.80	0.68	0.64	0.68	0.61	0.67	0.67
PatentBERT-GSAGE	0.73	0.84	0.89	0.72	0.60	0.67	0.60	0.67	0.68

Table 13: Accuracy for Citation classification (top vs bottom). We use Top 10% as the highly cited category (positive class) and Bottom 10% as the low cited category (negative class). Our graph-based methods often produce the best results (blue) and accuracy up to .89 indicating that they are effective in patent citation prediction.

Models	CPC Classes								
	A61			H04			G06		
	Citation Predictions @								
	3y	5y	10y	3y	5y	10y	3y	5y	10y
Doc2Vec-MLP	0.86	0.91	0.78	0.78	0.78	0.78	0.75	0.72	0.59
PatentBERT-MLP	0.83	0.92	0.90	0.79	0.79	0.79	0.79	0.73	0.65
Doc2Vec-GCN	0.87	0.91	0.78	0.83	0.83	0.83	0.79	0.72	0.59
Doc2Vec-GTN	0.83	0.89	0.77	0.78	0.61	0.50	0.69	0.72	0.60
Doc2Vec-GSAGE	0.85	0.90	0.79	0.81	0.59	0.51	0.75	0.73	0.61
PatentBERT-GCN	0.84	0.90	0.78	0.74	0.74	0.74	0.79	0.73	0.61
PatentBERT-GTN	0.84	0.91	0.56	0.81	0.63	0.56	0.72	0.73	0.64
PatentBERT-GSAGE	0.83	0.91	0.91	0.81	0.61	0.56	0.71	0.74	0.66

Table 14: **F1-Score of High Cited Patents:** We use Top 10% as the highly cited category (positive class) and Bottom 10% as the lowly cited category (negative class). Our graph-based methods often produce the best results.