Predictive Patentomics: Forecasting Innovation Success and Valuation with ChatGPT *

Stephen Yang†

November 2023

Abstract

Conventional approaches to analyzing structural data have historically limited our economic understanding of innovation. This paper pushes the boundaries, taking an LLM approach to patent analysis with the novel ChatGPT technology. I develop deep learning predictive models that incorporate OpenAI's textual embedding features to access complex, intricate information about the quality and impact of each invention. These models achieve an R-squared score of 42% predicting patent value, 23% for patent citations, and clearly isolate the worst and best applications. My techniques also enable a revision to the contemporary Kogan, Papanikolaou, Seru, and Stoffman (2017) valuation of patents with a median deviation of 1.5 times, accounting for potential institutional anticipation and generating substantial incremental value for economic applications. Furthermore, the application-based measures provide previously inaccessible latent information regarding corporate innovative productivity; a long-short portfolio based on predicted acceptance rates achieves significant abnormal returns of 3.3% annually. The models provide an opportunity to reinvent startup and small-firm corporate policy vis-à-vis patenting.

JEL Classification: G30, O32, O34.

Keywords: AI, ChatGPT, Large Language Model, Machine Learning, Innovation, Patents, Patent Success, Patent Applications, Patent Value, Textual Analysis, Natural Language Processing, FinTech.

---

*The author has benefited from helpful discussions with and suggestions from Vikas Agarwal, Lin William Cong, Rohan Ganduri, Jerry Hoberg, David Jerison, Wei Jiang, Kose John, Bing Liang, William Mann, Andrea Moro, Stefan Nagel, Hope Rogers, Fahad Saleh, Jing Xue, Baozhong Yang, and Alan Zhang, and comments and suggestions from participants in seminars and conferences at Georgia College & State University, the 2023 Academy of Behavioural Finance and Economics Conference and Theory Day, the CEBEX Conference on Behavioural Sciences, the FFEA Inaugural FinTech Conference, the Vietnam Symposium in Banking and Finance, and the Vietnam Symposium in Global Economic Issues. The usual disclaimer applies.

†Pace Academy, Email: stephen.yang25@paceacademy.org.

1. Introduction

Patents have been the subject of extensive study as the embodiment of innovation and Schumpeterian creative destruction. However, existing literature measuring the impact of patents is restricted to ex-post analysis. Specifically, a wealth of research examines the value and significance of patents, whether measured through market returns, citations, or other proxies (e.g., Hall, Jaffe, and Trajtenberg, 2005; Kogan, Papanikolaou, Seru, and Stoffman, 2017, henceforth KPSS). Additionally, despite recent advances in natural language processing, the technology has not existed to comprehensively analyze the content of patents at the level of human experts to complement the use of structural observations such as classifications or firm characteristics. This study fills the void by creating predictive models for patent value and application acceptance using the state-of-the-art ChatGPT technology. The ex-ante approach I take carries significant unexplored economic implications for innovating firms and investors.

This paper seeks to answer the following research questions. First, is it possible to accurately predict patent value and application acceptance ex-ante? Second, how can existing models concerning patent and innovation value be supplemented and improved using Large Language Models (LLMs)? That is, can qualitative, non-trivial information regarding the quality and disruptiveness of the innovation be extracted from the application text? Third, can these predictive models help companies and investors make better decisions apropos innovation?

I study a sample of roughly 900,000 applications and 2,000,000 patents from 2001 to 2020. I develop a deep learning model for patent acceptance that builds on the ChatGPT textual embedding, which captures information about the application text into a numerical feature vector. The embedding is supplemented with basic structural variables such as patent classification or firm size and fed into a neural network. LLM embeddings incorporate many qualitative aspects of textual information previously unavailable to researchers, including but not limited to sentiment, writing quality, technicality, and so on (Radford et al., 2018; Brown et al., 2020; Huang, Wang, and Yang, 2023). Given that patent applications are reviewed by subjective human readers, these textual variables and factors should be powerful predictors for the application's ultimate success or failure. My models include other innovations crucial for performance, such as a more modern activation function and nonlinear transformations of variables, discussed further in Section 3.

The sophisticated deep neural network model is the main predictor and achieves an F1 score of 86%.[1] The performance of this model is significantly superior to the benchmarks: *i)* a neural network without the embedding features and *ii)* another machine learning model (gradient boosting, a decision-tree-based statistical learning method) with the complete set of features. This demonstrates that both the textual embedding variables and the deep neural network structure are central to the primary model's superior performance. Especially important is that the principal model performs substantially better at isolating the "worst" and "best" applications. The difference in the success rates of the worst and best applications predicted by the main network and the benchmark model trained without the textual embedding is economically significant, circa 10%.

I render interpretability to these results via several distinct avenues. First, samples of the "worst" identified applications, where structural variables such as firm size and reputation alone would indicate a high chance of success, suggest the use of vague, repetitive language. Second, I examine the word clouds of the identified "worst" and "best" applications. There are major differences between the two groups, characterized by trends such as meaningless, vague, and overused language used in the worst applications. Importantly, there is also qualitative variation in the content and subject matter between the two groups; e.g., applications relating to solar panels are relegated to the worst group. The evidence thus suggests that the model is effectively capturing information about the topic of the innovation as well as writing quality and leveraging this data to make informed predictions. Third, I construct a proxy measure for the unobservable *application quality* by using the prediction of a separate neural network model trained solely with the ChatGPT embedding, such that the model only observes textual information as inputs. I demonstrate that applications filed by larger, older, and more experienced firms exhibit significantly higher *application quality* and thus experience a substantial advantage in the patenting process. This is consistent with hypotheses that the greater resources and experience available to industry leaders allow them to perfect their applications and apply inside knowledge of USPTO procedures to "game" the process.

I next apply a similar approach to predict the financial value of patents as per KPSS. Specifically, the prediction uses the same feature variables and neural network design as for acceptance. In this case, the primary model exhibits a performance of 42% adjusted R-squared for the full model,

[1] The F1-score is a metric of a prediction model's performance that balances the trade-off between avoiding both falsely identifying positive instances (precision) and missing positive instances (recall).

a substantial improvement of 24% over a benchmark model trained without the embedding. Additionally, a gradient boosting model trained for comparison on the complete feature set still outperforms the benchmark neural network by 10%. Together, these results prove that the ChatGPT textual embedding provides economically significant, previously inaccessible information.

Finally, I leverage similar techniques to predict patent citations, which are both economically and technologically relevant. The models use the same feature variables and neural network design as for value. An additional roadblock to predicting citations is the nature of the distribution, which is heavily "zero-inflated," meaning that most patents do not receive citations. I use an innovative dependent to circumvent this problem, which otherwise confounds all statistical learning methods. By predicting the *increase* in citations from a three- to ten-year horizon conditional on non-zero citations within the first three years, I am able to screen out many of the zero-observations while maintaining ex-ante validity. The results are strong, as the full model achieves an adjusted R-squared value of 20.1%. Most importantly, the benchmark neural network model trained without textual information only achieves a 7.9% adjusted R-squared, suggesting that the textual information supplied by ChatGPT is of utmost importance specifically for citations, possibly because of their scientific nature.

One potential concern with using the ChatGPT model is that it was trained using data up to 2021 and thus may exhibit "look-ahead bias" when applied to the analysis of earlier patents. This concern is, however, mostly alleviated by the nature of my use case. Particularly, I make use of the ChatGPT textual embedding vector as inputs for my deep learning model. The embedding vector is a general encoding of texts without any prompt or "focus" and is thus highly unlikely to contain granular, direct information related to patent success or value. In other words, one may intuitively consider that the signal-to-noise ratio ought to be very low regarding the potential incorporation of future information about the patent. To empirically demonstrate this, I consider an out-of-sample (OOS) test of the patent value prediction model in 2022, which generates very similar results to previous years. Reference Section 3.5 for further detail.

These new models carry significant economic implications. First, I use the predictive model for application acceptance to revise the KPSS estimation of patent value. KPSS introduce a scaling factor $1/(1-p)$ that accounts for the fact that investors expect a patent to be accepted with a probability p. They use a constant p_0 = 55% for all patents based on the argument that it is challenging to predict patent acceptance. I propose replacing the blanket constant scaling factor p_0 with the predicted chance of success $\hat{p}$ given by the machine learning model in this paper. Since the prediction from the embedding acts as an ex-ante simulation of the analysis of human experts or even natural language processing models employed by investors, the revised factor enables a potentially more realistic valuation of patents. The median proportional deviation of the new AI-based patent value measure from the KPSS value is 1.46 times, and the mean deviation is 2.65 times.

My estimator deviates most substantially from the KPSS measure in the outlier cases, i.e., very strong or weak applications for which the market reaction will be otherwise proportionally under or overestimated. Moreover, if the market does accurately judge the application's chances of success and acts efficiently, the potential undervaluation of other methods is nearly unbounded: incredibly disruptive and impactful inventions may be evaluated by the market as being near certain grants, and thus the reaction following acceptance will be extremely undersized. With a constant scaling factor, such patents may be misidentified as not valuable due to the limited reactions, while they are, in reality, very valuable. Specifically, for the patents deemed "best" by the predictive model, the assumption of a constant acceptance rate may undervalue by many times. However, it is worth mentioning that the average acceptance rate in my sample is 72.4%, higher than the 55% in KPSS's larger sample. Adjusting the KPSS scaling to this acceptance rate, the median deviation of the AI-based valuation from the adjusted KPSS valuation is still a sizable 51%, and the mean is 1.24 times.

To investigate whether my new AI-adjusted patent value measure contributes additional, economically significant information to the KPSS measure, I consider whether the new measure is associated with forward citations, following Kogan, Papanikolaou, Seru, and Stoffman (2017). Indeed, there is a strong positive association between forward citations and the new metric, with a higher level of significance than for the KPSS value. Additionally, when regressing citations on the metrics in a multilinear regression with both metrics, the AI-based metric remains more significant than the KPSS value. This is consistent with my finding that the AI-based scaling factor is orthogonal to the KPSS value and suggests that the new measure provides substantial incremental value.

Second, the models present the potential to manifestly enhance the patenting policies of companies. When companies' applications have a low predicted probability of acceptance based on the predictive model and are later accepted by the USPTO, I observe meaningful changes between the grant and application texts that significantly improve the predicted chance of success. However, while my findings are significant at the 1% level, the number of such active revisions of application texts is extremely small relative to the total number of weak applications, indicating that this strategy is under-utilized. This trend suggests that companies, especially startups and small firms that lack the expertise and resources to effectively "game" the system, can improve their innovation process by using machine learning models to screen and fine-tune applications before sending them to the USPTO.

The resulting outcome will be higher acceptance rates and more impactful innovation by these firms. This process could also encourage innovation and R&D investment, as the otherwise nebulous risk of failed applications threatens the loss of capital and revealing the new technology to competitors; risk-averse managers can act more confidently with the ability to level the "playing field" between small and large firms. A natural question left unanswered is: if an application is labeled as below average due to lack of expertise, how can the inexperienced inventor or firm revise it? I test the ability of ChatGPT to revise patent applications as a robo-advisor and find significant positive improvements in *application quality*. Although the AI cannot innovate by itself or create new ideas, its revisions significantly enhance the presentation of the application, boosting chances of success for the worst applications by 60% on average, proportionally. The theme of this finding is consistent with previous work in the literature which investigates the potential for "Man + Machine," i.e., the conjunction of human experts and AI (Costello, Down, and Mehta, 2020; Cao, Jiang, Wang, and Yang, 2022).

Furthermore, with the introduction of LLM technology to the market, it is increasingly important for firms to consider the reaction of institutional investors to textual information, as has been documented in the cases of corporate disclosure and the news (Cao, Jiang, Yang, and Zhang, 2023; Huang, Tan, and Wermers, 2020). This "feedback effect" should play an outsized role as patent texts are available for public access in bulk data, meaning that firms must monitor the impression they make on the models employed by investors. If nothing else, the use of machine learning models can allow firms to pre-emptively adjust and optimize for market reaction.

Third, the models provide new, ex-ante information regarding firms' latent innovative strength. I construct a firm-level measure of application strength based on the average predicted chance of success and the number of patent applications published in a certain month. This provides a good proxy for both the quality and quantity of the latent innovative strength of a firm, inaccessible through previous methods. Based on the predictions of endogenous growth models, this new measure should proxy for the future ability of the firm to succeed in innovation (i.e., patenting), an important factor in market returns (Klette and Kortum, 2004). Indeed, a long-short portfolio constructed based on application strength achieves statistically significant yearly abnormal returns of 3.3%.

This paper makes several important contributions to the literature. First, it makes major methodological contributions, from the advanced ChatGPT technology to multiple innovative approaches to the secondary machine learning step. These sophisticated methods are critical to achieving the superior performance of this paper's models. Second, several of the economic implications that I demonstrate, particularly the revised measure of patent value, are of importance to several fields of the literature, from corporate policy regarding innovation to studies of endogenous growth to market anticipation, and I am among the first to study *patent application acceptances* extensively. Finally, I try to break open the machine learning "black box" by taking several novel approaches to the interpretation of the models. The interpretability of machine learning models and applications is greatly important economically, especially in the case of complex and powerful models such as the textual embedding of LLMs like ChatGPT.

Moreover, my study relates to several particular strands of research. First are the studies of the valuation of innovation. Determining the value of innovation is vital to both the market and individual firms, as innovation has an outsized impact both socially and for growth (Drucker, 2014; Bloom, Schankerman, and Van Reenen, 2013; Kogan et al., 2017). There is a long history of literature studying patent valuation. Some of the earliest works directly examined the relationship between patent release and stock returns, finding evidence that patent grants are highly correlated with market returns (Pakes, 1985; Austin, 1993). Hall, Jaffe, and Trajtenberg (2005) use raw patent citation count instead as a proxy for innovative success and find a strong correlation with the firm's market valuation. KPSS develop a new measure based on market reaction to the announcement of patent success. Kline, Petkova, Williams, and Zidar (2019) extend the KPSS ex-ante measure to include non-public firms and analyze both accepted and rejected patents, as well as worker productivity and wages. This paper also builds on the KPSS value, scaling based on the prediction of application success, which better proxies the actual investor evaluation of the innovation. This method has the potential to help companies and investors to better recognize and distinguish the most impactful innovations, a well-documented problem (Cohen, Diether, and Malloy, 2013; Hirshleifer, Hsu, and Li, 2013; Fitzgerald, Balsmeier, Fleming, and Manso, 2021).

The second strand is textual analysis in the studies of innovation. Kelly, Papanikolaou, Seru, and Taddy (2021) also take a machine learning approach to patent evaluation but instead measure technological disruptiveness. Bowen, Frésard, and Hoberg (2023) investigate patent texts to create a measure of novelty/stability and find that startups in rapidly advancing fields exit via IPO more frequently. These measures of disruptiveness and novelty are correlated with, but not a direct proxy for, the economic value of a patent (as other factors such as creative destruction and commercialization often take a more direct role than genuine scientific inventiveness) that this paper focuses on. Another innovation paper that leverages machine learning techniques in the context of patenting is Zheng (2022). They approach the questions from the point of view of the regulatory body, i.e., the USPTO, and theorize the potential use of machine learning as a robo-advisor to assist patent application reviewers while highlighting flaws in the current, subjective system. They find that the implementation of machine learning algorithms, in the patent review process, has the potential to solve problems of false acceptances (which lead to litigation and poor performance) as well as false rejections (which are inevitably harmful to the slighted firm), and, as a result, failures in venture capital (VC) funding and initial public offerings (IPOs). The "screening and revision" sections of this paper differ in that I approach the question of application screening from the perspective of the innovating firm. That is, my theorized process allows firms to maximize the chances of acceptance for individual applications, within the existing system of patent examination. Biasi and Ma (2022) consider the question of innovation from the scientific perspective, analyzing the gap between academic articles and university education using NLP.

Third, my study speaks to the literature on textual machine learning and natural language processing applications in finance. For the last decade, researchers have used methods from Word2Vec up to BERT in many situations (Hanley and Hoberg, 2019; Cong, Liang, and Zhang, 2020; Li, Mai, Shen, and Yan, 2021; Acikalin, Caskurlu, Hoberg, and Phillips, 2022; Bybee, Kelly, Manela, and Xiu, 2023; Cao, Jiang, Yang, and Zhang, 2023). With the introduction of ChatGPT and related large language models, several studies in finance and management writ large have begun implementing this new wave of LLM technology (Kim, Muhn, and Nikolaev, 2023; Hansen and Kazinnik, 2023; Jha, Qian, Weber, and Yang, 2023). This paper applies the state-of-the-art embedding technology by OpenAI's ChatGPT, which encodes richer information and higher-level concepts than previous models. The superior predictions achieved by the models indicate that the emergence of sophisticated LLMs has far-ranging implications for finance more broadly.

Finally, my work contributes to the finance literature that builds machine learning models and renders interpretability to them, e.g., Freyberger, Neuhierl, and Weber (2020), Gu, Kelly, and Xiu (2020), Cong, Tang, Wang, and Zhang (2021), Cong, Feng, He, and He (2023), and Chen, Pelger, and Zhu (2023). My paper adds to prior studies by constructing sophisticated deep-learning models that incorporate LLMs and providing innovative, interpretable analyses.

The remainder of the paper is organized as follows: Section 2 outlines the data sources, sample, and variables used. Section 3 describes the machine learning techniques used to predict application acceptance, patent value, and citations, as well as benchmark tests and interpretability. Multiple economic implications in connection to the literature are presented in Section 4. Section 5 concludes.

2. Data and Variables

2.1. Data Sources and Sample

The primary data source is the USPTO's PatentsView service, which provides complete documentation and texts for all patents from 1976 and applications from 2001 in the United States. This official data set is more complete and comprehensive than previously available methods of obtaining data. This is supplemented by market information from CRSP/Compustat provided by the WRDS service. I use the entire sample of patents and applications, contingent on the assignee having a valid "permno" to link to market data.[2] Patents and applications with multiple assignees are excluded to avoid duplication. The final sample contains 855,891 applications and 3,177,942 granted patents. The disparity between the two is partly a result of the USPTO not publishing application texts prior to 2001. I thus truncate the patent dataset, in most tests, at 2001, reducing the patent sample size to 2,239,148. The remaining difference is primarily a result of applications filed

[2] The assignee names must first be cleaned and standardized extensively to match with "permno" databases. Additionally, visual inspection is used in cases of very close but not absolute matches between firm names post-cleaning. The problem of the inconsistent application assignee data is discussed in KPSS, who elect to avoid patent application data and associated ex-ante tests.

before 2001 as well as missing assignee data in the application files, i.e., the USPTO data lacks valid company assignees for a large portion of applications in addition to the inconsistency in formatting.

I use a rolling window training sample; all models are trained on data from the three preceding years, e.g., the predictive model for application success in 2004 is trained on all applications from 2001 to 2003. This ensures that all trends captured by the model will still be contextually relevant and avoids the problem of using unavailable information for prediction while maintaining a sufficiently large training sample for deep learning models. Finally, the application data is truncated by nature, as evaluation can often take several years. Thus both 2021 and 2022 are excluded from the application sample. In 2020, the truncation effect is in theory less prominent, but as the pandemic introduced new trends and factors not present in training data, it is also excluded. As a result, the effective test sample for applications ranges from 2004 to 2019. Figure 1 shows the time-series variation in the average acceptance rates for applications throughout the sample period. First, I observe substantial variation from less than 70% to over 80% in rates. The acceptance rate is the lowest before and during 2008, coinciding with the housing financial crisis. A potential mechanism behind this trend is the fact that firms often reduce funding in innovative research during times of financial friction, and particularly the 2008 crisis (Mezzanotti and Simcoe, 2023). This suggests that, although still attempting to patent, firms produced overall lower quality inventions, and thus acceptance rates plummeted.

[Insert Figure 1 Here]

2.2. Variables

My primary independent variable is the embedding vector generated by the GPT model, obtained from OpenAI's publicly available API. The title and abstract of all applications and patents are combined and fed through the model to obtain the embeddings. Each embedding vector is a list of 1,536 numbers, which contains many dimensions of information about the text (e.g., sentiment, topic, et cetera). Fundamental structural variables supplement the embedding, which is critical to provide context, such as firm size and industry, to any given application or patent. Specifically, I use the Cooperative Patent Classification (CPC) system (i.e., Class A, human necessities; B, operations; C, chemistry; and so on), a USPTO-generated measure of "AI" patents,[3]

[3]The AI classification published by the USPTO is only available for patent grants, not for patent applications.

separate classes of Information and Communication Technologies (ICT), biotechnology, and high tech, which I generate based on the classification by the European Union,[4] number of CPC classes, the natural logarithm of the number of claims, whether an assignee is a research institution, and the Fama-French 12 industry classification, obtained from Kenneth French's website.[5] For application success prediction, the natural logarithm of market capitalization is also included. The market capitalization is calculated in the nearest quarter prior to publication as the product of shares outstanding and price, and adjusted for inflation.

The dependent for success prediction is a simple binary dummy, equal to one if the patent is granted and zero if it is rejected. I use the KPSS measure for patent value, calculated based on the market reaction to the announcement of patent acceptance in a three-day window per the procedure in the 2017 paper. The patent citation dependent is the increase in citations from a three- to ten-year horizon, while the models are trained and tests conducted on the subsample of patents that receive at least one citation within three years from publication. Section 3.4 discusses patent citations in further detail.

3. Machine Learning Predictive Models

3.1. Neural Networks

The availability of OpenAI's ChatGPT API to the general public, specifically the release of the Ada-002 LLM embedding model on December 15, 2022, gives researchers unprecedented access to deep learning capabilities. Ada-002 is OpenAI's top-of-the-line embedding model, capturing as much as four times the context of previous models, and provides an opportunity to utilize nuances of textual information previously inaccessible or impractical to obtain due to labor constraints. This technology is a significant improvement, as previous methods such as Word2Vec fail to account for context, thus losing the meaning from phrases or conditional modifiers as well as only being useful for basic tasks such as sentiment analysis which do not require recognition of higher-level ideas. Ada-002 is a transformer model, which means it uses "attention" vectors to ultimately output an embedding that transforms an entire body of text into a vector of 1,536 real numbers. The resulting embedding vector captures the meaning and nature of the text in its entirety rather than in disjoint

[4]European Union. 2006. Eurostat indicators on high-tech industry and knowledge – intensive services.
[5]Available at https://mba.tuck.dartmouth.edu/pages/faculty/ken.french/data_library.html.

parts. My employment of this technique joins the other first applications of transformer technology in the finance literature, such as Cong et al. (2021), Huang, Wang, and Yang (2023), and others.

This embedding vector then has many applications, from being used in a generative model such as ChatGPT, to determination of document similarity, searching, classification (such as review sentiment analysis, a popular benchmark in the neural network community), and even unsupervised clustering problems. This paper uses the vector as a feature (with 1,536 components) in a secondary machine learning approach to predict both the value of patents and the success of applications. Given that previous literature regarding patents leverages earlier word embedding methods, such as GLoVE or Word2Vec (e.g., (Cong, Liang, and Zhang, 2020; Xiao, Wang, and Zuo, 2018), and in finance generally, only earlier, less powerful transformer models such as BERT (e.g., Huang, Wang, and Yang, 2023; Cao et al., 2023), this is a substantial step forward and provides potential for application to other fields. It is likely possible to achieve even better results, particularly for patent value, by including more variables from the corporate side, as the models in this paper only utilize fundamental variables such as firm size and patent class in order to demonstrate better the significance and predictive power of the embedding features.

It is important to note that the embedding approach I take differs significantly from another ChatGPT-driven technique. Particularly, one may query ChatGPT directly, as with normal day-to-day use, with text and question(s). This method is employed by Jha, Qian, Weber, and Yang (2023), Kim, Muhn, and Nikolaev (2023), and Hansen and Kazinnik (2023), and provides unique channels to interpretability as well as potential for a wide range of applications. On the other hand, the embedding can be thought of as the back-end processing used for the consumer ChatGPT model, as it comprises the entire model except for the final generative step, instead producing a numerical representation of the initial text given which I then use as input for my own neural network models. While this approach may exhibit certain drawbacks, the embedding provides greater potential for highly specific applications that ChatGPT may not have the capacity to answer, such as in the case of patent value and application acceptances. Furthermore, the difference is significant for tests conducted in Sections 3.2, 3.5, and 4.1 and is discussed further where relevant.

For both predictive models, I feed the embedding vector together with the structural variables as the feature vector into a three-layer feed-forward neural network, or multi-layer perceptron (MLP). OLS and similar linear-based methods are unsuitable for this situation because of the high dimensionality of, and complex interactions within, the embedding. The choice of the activation function, which regulates the transmission of information signals between "neural nodes" in the network, is often overlooked in economic applications of neural networks. However, the choice of function is in fact perhaps the most crucial hyperparameter (besides layer design). Most finance and economic studies use the simple ReLU activation function, which equals 0 if x is negative, and x otherwise. Some other traditional functions are tanh, softplus ($\ln(1 + e^x)$), and *ELU* ($e^x - 1$ for negative x, and x for positive x). I utilize a new activation function, *Mish*, defined as in Misra (2019):

$$Mish(x) = x \cdot \tanh(\ln(1 + e^x)) \quad (1)$$

Mish is a novel addition to the traditional family of activation functions. While *ELU* or even *ReLU* can perform reasonably well in cases of low dimensionality, such as when the embedding vector is not included, I find *Mish* to perform significantly better in cases with large sample sizes and feature counts, likely because of a combination of its self-regularizing nature and its ability to handle negative inputs and to capture highly non-linear and irregular relations between the embedding features. The invention of the *Mish* formula is a major innovation and provides a significant improvement in performance over traditional *ReLU* activation (Misra, 2019). *Swish* is another contemporary activation function, defined as $x \cdot sigmoid(\beta x)$, where β is learned (Ramachandran, Zoph, and Le, 2017). Although *Mish* and *Swish* appear similar, there are differences in their respective first and second derivatives which change how the models evolve (in testing, *Swish* fails to match even *ReLU*). Several of the above-discussed activation functions are graphed in Figure 2. More implementation details of the deep learning model are provided in Appendix A.

[Include Figure 2 Here]

3.2. Predicting Acceptance

3.2.1 Main Model and Benchmarks

These methods are first applied to predict the likelihood of application acceptance. I treat this as a binary classification problem, with the dependent variable *Accepted* equal to 1 if the application becomes a granted patent and 0 otherwise. Three models are trained and tested. First, the primary deep learning model is given the complete list of features available for patent applications. The dependent *Accepted* is heavily skewed by nature, as the USPTO accepts roughly 70% of patent applications. Thus, the model is optimized for binary cross-entropy (log-loss) to best adapt to the skewness, which is defined as

$$\textit{Binary Cross Entropy}(\{y_i\}) = -\frac{1}{N}\sum_{i=1}^{N}\left(y_i \cdot \log p(y_i) + (1-y_i)\cdot \log\left(1-p(y_i)\right),\right) \tag{2}$$

where $p(y_i)$ represents the probability of one, and $1-p(y_i)$ the probability of zero.[6] Second, I benchmark by training a similar deep learning model without the embedding features to determine the incremental improvement generated from the textual information. Third, I test the gradient boosting model (a commonly used machine learning method relying on decision trees) on the comprehensive set of variables for comparison. In untabulated results, a linear support vector machine classifier (SVC) is trained and fails to outperform the null predictor (predicting success for every application), potentially due to the high dimensionality of the features.

Five statistics, commonly employed for the evaluation of binary classification models, are presented in Table 1, illustrating the year-by-year performance of the primary deep learning model. AUC is the area under the Receiver Operating Characteristic (ROC) curve and measures the probability of ranking a true positive higher than a true negative. Accuracy is the overall percentage of correct predictions. Precision is the percentage of true positives in all labeled positives, and Recall is the percentage of true positives labeled as positives. F1 Score is the harmonic mean of Precision and Recall and, in Bayesian terms, typically provides a good balance between optimizing against type 1 and type 2 error. The model performs well in all measures, with an average Accuracy of 77%, and an average F1 Score of 86%. Importantly, it achieves an AUC score of 57%, a Precision of 79%, and a Recall of 95%, averaged across all years. AUC is the best criterion for evaluation and identifying the best model in the specific case of this paper due to the aforementioned skewness of the sample. For example, the null predictor that always predicts acceptance would have an accuracy of more than 70%, but an AUC of only 50%, the same as a random predictor.

[Insert Table 1 Here]

Second, Table 2 shows the comparison of the main model with both benchmarks. While the

[6] Optimizing for Accuracy or F1 Score biases the model towards recreating the null predictor.

benchmarks achieve deceptively similar results in Accuracy and F1 score, both fall short in AUC, demonstrating that the conjunction of the ChatGPT embedding and advanced deep learning techniques is the key to consistently distinguishing the bad applications from the mediocre or good. This is again demonstrated by the excessively high Recall (true positives correctly classified) of both benchmarks. Nearing 100% Recall suggests that both models approach the null predictor, as it is otherwise impossible to avoid false negatives completely (without an accuracy of 100%).

[Insert Table 2 Here]

3.2.2 *Distinguishing Worst vs. Best Applications*

I qualitatively demonstrate this difference in AUC between the principal model and the benchmark without the embedding features by evaluating the predicted "worst" and "best" applications from each year in the sample for both models. The results are presented in Table 3. The fully trained model performs exceptionally well at both the highest and lowest end of prediction. For example, the yearly top 100 predicted applications have a 96% acceptance rate, and the bottom 100 only 26%. Even enlarging the yearly cutoff to 1000, the model still maintains a success rate of 94% for top applications and 34% for the bottom. The ability to accurately predict the worst applications is surprising because the data is skewed so heavily toward acceptance. In comparison, the benchmark reaches a comparatively weak 87% rate for the top 100 yearly predictions and 37% for the bottom 100. Clearly, the primary model is better at picking out the applications likely to succeed and those likely to be denied. Thus, the benchmark is significantly less valuable economically, as it neither provides the ability to accurately distinguish the best applications to invest in, nor the worst to screen out; these are only possible with the additional textual information provided by the embedding.

[Insert Table 3 Here]

3.2.3 *Interpretations of the Model*

Interpretation is both significant and challenging in the case of complex neural network models. As opposed to other approaches that directly ask ChatGPT or other generative models their questions, in which case one may append "give the justification for your answer," the embedding model acts as a "black box" in nature, taking the input text and outputting an embedding vector; there is no opportunity to "ask" the model about its choices, no other output. Although there are no consistent, generally agreed-upon tests for interpreting neural network predictions of this type en masse, I develop and utilize several intuitive methods for interpretation.

First, visual investigation finds consistent trends in some of the worst applications in the sample, which may be isolated by the ChatGPT textual embedding and subsequently incorporated in the predictive model. Appendix B shows the titles and abstracts of some of the worst applications per the prediction of the model. Exhibit 1 is a Class A (human necessities) patent application filed by a leader in the pharmaceutical industry. Given these structural variables, human analysts or the casual observer would likely predict a high chance of success. After all, this is a case of a market leader in the drug industry applying for a patent for a newly developed drug. However, the model is highly critical and correctly predicts that it will not be accepted, assigning a meager chance of 5.9%. This indicates that it must have extracted additional information from the text of the abstract and title. The text is unclear regarding the purpose or innovation of the invention is (a new synthesis, delivery method, et cetera). Notably, the abstract also repeats the same sentence twice ("the present invention relates to pharmaceutical compositions…") with a change in a chemical name, a perplexing choice that makes the application nearly unreadable. The predictive model potentially accounts for the subpar, unclear writing in its assessment of the application.

Exhibit 2 is also a Class A application, filed by a different leader in the pharmaceutical industry. In this case, the title of "Method 741" is extraordinarily unhelpful and the abstract comprises a single sentence fragment. Exhibit 3 is also Class A (the overall lower quality of biotech patents results in an abundance of Class A applications among the worst in the sample), filed by a less prominent but still valuable pharmaceutical company. Of note is that the abstract claims to have found a cure for obesity, anxiety, PTSD, ADHD, Tourette's, and others through a single drug. Additionally, the abstract delves into excessive detail, listing over a dozen chemicals as components of the drug in one section (granular facts that should be left to the detailed description of the patent). Exhibit 4 is again a pharmaceutical application. It seems to repeat the same sentence four times, substituting, respectively, the phrases "treating a patient," "providing a pharmaceutical composition," "add-on therapy," and "use… in the preparation of a combination." There seems to be no discernable or meaningful difference between these claims. Also of note is that abstracts mentioning the chemicals "laquinimod" and "pridopidine" appear frequently in the "worst" identified applications and are also often rejected. This indicates that the model likely extracts and considers non-trivial information about the content of the innovations as opposed to only writing style or quality. Finally, Exhibit 5 is a Class G (Physics) application filed by a leading computing hardware company. It follows a similar pattern to the other applications, with three repetitive, vague sentences. Although all three seem to convey the same meaning, the exact nature and importance of the application are somehow still lost on the reader.

Next, in order to further interpret the model's predictions and identify the distinguishing factors between the "best" and "worst" applications, I create visual representations of the texts through word clouds. Specifically, I first isolate the 30,000 best and 30,000 worst applications based on the full model predicted chance of success from the entire sample, then split based on patent classification (CPC and generated). Figure 3 shows the word clouds for the worst and best applications in climate, biotechnology, and ICT, while the remainder are reported in Figure IA.1, found in the Internet Appendix. Importantly, there are significant differences between the worst and best applications within every classification and some consistent trends across all groups. First, the words "plurality," "portion," and "may" generally dominate the best applications and are hardly found in the worst. Second, with the exception of climate applications (CPC class Y), the terms "data," "user," and "information" appear much more frequently in the worst applications. For all classifications, phrases such as "the present invention," "invention provides," and so on dominate the worst applications. These trends indicate that the model makes qualitative assessments of the writing quality of applications, as the worst demonstrate characteristics of repetitiveness and vague/useless ("invention," "information," et cetera) language.

Additionally, there are also major differences in content and subject matter. For example, technology such as "solar cells" and "photovoltaic *[technology]*" are identified by the model as significantly less likely to be accepted. This could potentially be due to an oversaturation of solar panel (photovoltaic) technology, leading to the denial of most applications by the USPTO under the "originality" grounds. In the best applications, innovations related to printers often appear, such as "ink jet," "ink," "change ink," and so on. In the area of biotechnology, the hardest area to receive patent grants in by a large margin, the best applications tend to be about "radiation," "x-ray," "image," "sensor," and so on, while the worst tend to propose "pharmaceutical composition(s)" and often involve "nucleic acid." A litany of interpretable differences such as these in the word clouds lends credence to the claim that the embedding and model in conjunction are able to identify trends in acceptance based on both writing quality/style as well as subject matter/qualitative content of the applications.

[Insert Figure 3 Here]

3.2.4 Economic Factors and Application Quality

Given these trends, I next investigate whether economic and corporate factors drive these phenomena of poor application writing. Particularly, do older and more experienced firms produce more polished applications, given their detailed knowledge of USPTO processes and substantial R&D resources? Or, do the largest firms submit too many applications to carefully revise each and every one, prioritizing aggregate patent count over the quality of individual applications? The answer is not immediately clear, although intuitively, one may expect that the first is more likely to be true. In order to answer this question, I first isolate the effect of writing on firm success by training a distinct neural network to predict application success with only the embedding as input. The resulting prediction proxies directly for the unobservable and highly impactful *application quality*, as the model only factors in the text itself. Figure 4 provides a litmus test for the validity of this measure and the accuracy of the model. Panel A documents the acceptance rate across different patent classifications (CPC and generated) and years, while Panel B shows the same but for *application quality*. Many of the prevailing trends in acceptance rate are also present in the measure of *application quality*, such as the extremely poor performance of biotechnology applications, the strong performance of applications in electronics, engineering, and ICT, and a gradual increase in the rates for most but not all classifications. Given that the model is not trained with these characteristics as input, the persistence of these trends suggests that the model is accurate and identifies economically interpretable trends in acceptance.

[Insert Figure 4 Here]

I then estimate the following regression, for firm-application-time (i, j, t):

$$\begin{aligned} \textit{Application Quality}_{i,j,t} &= \beta_1 \textit{Size}_{i,t} + \beta_2 \textit{Age}_{i,t} + \beta_3 \textit{Application Stock}_{i,t} \\ &\quad + \gamma \cdot Controls_{j,t} + \delta_i + \epsilon_{i,j,t} \end{aligned} \tag{3}$$

Firm size is the natural log of shares outstanding times price, adjusted for inflation, and age is the number of years since the firm first appeared in CRSP. Application stock is defined as the cumulative number of patent applications (regardless of success) filed by a firm before the month of the current application. The results of the regression are shown in Table 4. Additional specifications are also tested with each independent alone with firm-fixed effects. Year-fixed effects are not suitable for this situation, as combined with firm-fixed effects, the controls would completely absorb firm age. I find strong positive associations with *application quality* from all three independents, firm size, age, and application stock, all significant at the 1% level. Overall, this demonstrates that a firm's experience and resources significantly affect the quality of its applications.

It is important to note that these aggregate regression results are not inconsistent with the granular examples of individual applications provided above but rather strengthen the claim that the textual embedding is capturing high-level concepts beyond those available from structural variables. Given the positive correlation between firm size and *application quality*, a model trained only on structural factors will always label an application from an industry leader as better than that from a smaller firm, given all else equal. However, given the sheer number of applications filed by these industry leaders, the largest firms should logically also submit some of the worst applications. That the predictive model is able to overcome "bias" in the data towards these large firms and correctly discern when their applications are lackluster indicates the value of the textual approach.

[Insert Table 4 Here]

3.3. Predicting Value

Next, I predict the KPSS measure of economic value, determined via market response to the publication.[7,8] I follow the same procedure as for predicting application acceptance to create the sample and train the model with a rolling window. The dependent variable is the KPSS patent value scaled by market capitalization, calculated in the quarter prior to publication.[9] However, direct training fails to produce good results, as patent value is highly skewed by nature even after the adjustment. As statistical learning techniques in general and especially neural network models perform best on demeaned dependents with lower variance and skewness, I transform the dependent variable before training and testing the models. I test several normalization methods in pre-processing, including Box-Cox, quantile normal, $\ln(1+y)$, and simple z-score normalization (standardization). Of these, Box-Cox, a method introduced in Box and Cox (1964), performs the best. This method is commonly used in the financial literature to perform "Box-Cox Regression," in which OLS is applied to a dependent which has been put through the Box-Cox transformation (e.g., Bhagat and Frost, 1986). In cases of highly skewed positively valued dependent variables, applying Box-Cox significantly improves the performance of OLS. However, machine learning papers in finance and economics generally apply the more basic $\ln(1+y)$, standardization, or no transformation to the dependent when training a neural network. The application of Box-Cox in the models of this paper is thus an innovation, as its use significantly improves predictive power in terms of R-squared while still maintaining applicability as relative ranks are unchanged. The transformation is calculated as follows:

$$y_i^{\lambda} = \begin{cases} \frac{y_i^{\lambda}-1}{\lambda} & \text{if } \lambda \neq 0 \\ \ln(y_i) & \text{if } \lambda = 0 \end{cases} \tag{4}$$

Note that the use of Box-Cox calculates λ in sample (traditionally, to maximize log-likelihood), which could invalidate statistical results. I thus calculate λ only on the training set and apply the

[7]The textual embedding is, by nature, directly correlated with the revised valuation presented in section 4.2 and thus would be an overly powerful predictor. Any tests performed on the revised valuation would not provide a good objective metric for quantifying raw performance or determining the significance of the embedding.

[8]The specific formula is provided in KPSS. The code to construct the measure is publicly available at https://github.com/KPSS2017.

[9]The scaling by market capitalization removes the influence of firm size on patent value and makes its distribution less skewed.

same transformation to the test sample, which has no discernible effect compared to applying the Box-Cox transformation to the entire sample. With the Box-Cox-transformed KPSS value as the dependent, I again train a neural network model with 3 intermediate layers, as well as the benchmark model without embedding and the gradient boosting model for comparison.

The performance of the principal model's predictions is reported in Table 5. It achieves a strong adjusted R-squared score of 42%, with a trend of increasing performance with time from 37% in 2004 to 48% in 2020. The improvement is potentially a result of better market incorporation of patent analysis in recent years, fueled by institutional investors' use of machine learning techniques. If true, these trends would mean that the market better accounts for the genuine value of each individual patent.

[Insert Table 5 Here]

The primary model is then compared with both benchmarks in Table 6. First, comparing the full and no embedding models, there is a significant 25% improvement in adjusted R-squared when adding the embedding to the feature matrix, even using the best machine learning methods for both tests. The considerable increase proves that the ChatGPT-based textual embedding vector is a powerful and non-trivial predictor for value even when paired with conventional structural variables. Second, the gradient boosting model is unable to match the neural network in performance, reaching only 27% adjusted R-squared with the full list of features. This is 16% weaker than the full model, and the disparity demonstrates the superiority of the advanced machine learning techniques over the gradient boosting model. However, even gradient boosting still outperforms the advanced neural network trained without the embedding by 10%, further indicating the predictive power and importance of the textual embedding. Thus, the conjunction of both the adoption of the neural network model and the inclusion of the textual embedding features is the most important, although the embedding seems more powerful.

[Insert Table 6 Here]

3.4. Predicting Citations

In addition to economic value, an important measure of the significance of patents is citation count. Analogous to academic papers, the number of references a patent receives may indicate the impact it has on the R&D industry, the level of its scientific sophistication and innovation, or even its potential for licensing. The existing literature studies the importance of patent citations from varying economic and scientific angles (Hall, Jaffe, and Trajtenberg, 2005; Lanjouw and Schankerman, 2001; Abrams, Akcigit, and Grennan, 2019). The contribution of my paper, in this case, is again the conjunction of an ex-ante approach and the power of the ChatGPT technology. I apply a similar approach to predicting citations as with KPSS value, again using neural networks and leveraging the *Mish* activation function and Box-Cox transformation.

The major roadblock to successfully predicting citations is the ill-behaved nature of the distribution of the data; over 70% of all patents in the sample receive no citations. This phenomenon is known as a zero-inflated distribution and confounds almost all statistical learning techniques, including advanced machine learning. Note that the problems that this trend in the data presents are mostly unique to the ex-ante approach, as there is no way to know which patents will not be cited. One potential way around this problem involves using a "two-step" model, where the first model acts as a classifier, "filtering" out the zero-citation observations. However, this approach fails to produce results in the patent dataset, likely because there is not much difference between patents that receive no citations and those that receive one or two.[10] Therefore, I instead take an approach reminiscent of time-series analysis to solve the zero-inflation problem. First, the dataset is filtered for all patents that do not receive citations within a three-year horizon from publication, removing the technologically and economically insignificant patents. Next, the model is trained to predict the increase in citations from the first three years to the first ten years after publication. This method produces a substantially more powerful predictive model, and although it cannot be applied to patent applications or pre-grant patents, long-term citations are equally important in terms of value to the company and as a measure of scientific significance. This approach requires the dataset to be truncated at 2012, as later patents have not yet been published for 10 years. Even with this truncation, the sample and training window sizes remain sufficiently large.

Table 7 reports the results of the primary model for citations, trained on all data and using the best machine learning method. While the performance, in terms of R-squared, is weaker than that of the economic value models, it is still definitively significant and perhaps more impressive given

[10]In fact, conditional on that the zero-observations are removed (i.e., directly training on the subsample of non-zero citation patents), the model is able to predict citations accurately. Regardless, this is moot as even advanced classifiers are unable to successfully separate the zero-citation and nonzero observations.

the confounding nature of the dataset. Specifically, the model achieves an average performance of 20.1% in adjusted R-squared across the entire sample, a relatively powerful result. An interesting, but difficult-to-explain, trend in performance is that the model performs substantially worse in the years following 2008 than those prior, dropping from an average of 22.8% to 18.0%. The most logical hypothesis is that this phenomenon is related to the aforementioned anomalous patenting behavior following the financial crisis, observed in aggregate acceptance rates and application count, although this paper leaves the identification of a specific mechanism for future research.

[Insert Table 7 Here]

Table 8 reports the comparison in performance of the primary model with the benchmarks. As with the other two applications, the main model is benchmarked against a neural network trained without textual data, and a gradient boosting model trained with all input variables. The most striking result is the incredibly poor relative performance of the no-embedding neural network. Particularly, the mean performance is only 7.9%, less than half that of the full model. This proportional difference is much more dramatic than that observed in the cases of application acceptance and patent economic value. On the other hand, the gradient boosting model achieves results much closer to the primary model with a mean result of 14.6%, although still definitively weaker than the advanced neural network. The most important takeaway is that the improvement in performance driven by the *textual embedding* is more robust than that found in the models for patent value, which further strengthens the hypothesis that the textual information provided by ChatGPT is of utmost importance for prediction. This is intuitively sound, given the economic and technological importance of the actual content of the patent. Given this, it is fair to say that models without the use of this groundbreaking technology are mostly unable to accurately predict citations. A potential driving mechanism for this larger difference is that citations may be determined primarily by the specifics of the patent rather than broader, structural factors such as industry, although it is difficult to test such a hypothesis. It is also possible that the difference is merely a result of the complexity and ill-behaved nature of the citation data, even after resolving the zero-inflation problem – perhaps a higher-dimensional approach is necessary to capture complex trends in the distribution.

[Insert Table 8 Here]

3.5. Look-ahead Bias

An important issue to address for the application of GPT and other LLM models is the potential for look-ahead bias when studying historical data. Specifically, the GPT 3.5 and GPT 4 models are trained on a corpus of text ranging up to 2021. This means that results achieved by any models on textual data prior to 2021 are potentially invalid statistically. In the case of this paper, there are two ameliorating factors that lend credence to the results despite this potential issue.

First, I use the GPT *embedding* rather than the generative model. While this is a component of the full ChatGPT model, its purpose is only to convert a section of text into a vector of numbers. Consider that, if one queries the GPT model directly with context, the "attention" of the model, with some substantial abuse of the term, is focused on a subset of its knowledge, for example, specifically relating to innovation and patenting. However, the embedding is not given such a prompt or area, and thus uses its entire knowledge base. Additionally, the embedding is designed to incorporate *all* aspects of the text, primarily focusing on writing, diction, sentiment, and more. Taken together, these facts suggest that the signal-to-noise ratio of any potential look-ahead bias is prohibitively low (although this is not strictly measurable or, thus, testable), such that it ought not to invalidate statistical significance results. Particularly, it would be fair for one to suggest that it would be rather unreasonable to claim that the embedding model is using future information about the evolution of technology and encoding the value of a certain patent within the numerical vector it produces. Additionally, while the embedding is trained on information leading up to 2021, the actual predictor model is trained on the three-year rolling window sample. This means that any trends that the model incorporates for prediction must be present in the three years prior to the test year, which also resolves part, if not all, of the concerns.

Second, in order to scientifically verify the hypothesis that the bias ought to be minimal in the case of this paper, I conduct a test of out-of-sample (OOS) performance for value prediction. Although it is not feasible to conduct such OOS tests for application success or citations (as the data for 2022 is severely truncated for both by the nature of the dependent variable), persistence in the performance in the value model as well as the aforementioned factors specific to my use of the embedding would suggest that similar consistency should be present in the cases of the other two dependent variables. Table IA.1, given in the Internet Appendix, shows that the performance of the primary model for patent value prediction in 2022 is 42.4%, similar to other years. Furthermore, there remain substantial gaps between the main model and benchmark models. These results empirically resolve most remaining concerns regarding look-ahead bias.

4. Economic Implications of Forecasting Innovation Success

The successful prediction of application success has several economic implications in terms of corporate patenting policies, measurement of patent value, and latent value of patent applications, which I explore in this section.

4.1. Screening and Revision of Patent Applications

The Supreme Court wrote in 1892 that "the specification and claims of a patent… constitute one of the most difficult legal instruments to draw with accuracy."[11] Thus, ways to screen and improve the quality of patent drafting are potentially of great economic importance. Given a good interpreter of applications and an effective predictive model, innovating firms should be able to maximize their chances of patent success by running prospective applications through the model before submission. This augmentation is of outsized importance to small firms and start-ups, for whom the company's survival may very well hinge on the acceptance of a single critical patent. Without the resources to hire patent lawyers that industry leaders such as Apple employ on a daily basis, these companies often produce weaker application texts and lack inside knowledge of how to "game" USPTO procedures. Specifically, Section 3.2 reports my findings that, controlling for fixed effects, smaller and less experienced (proxied through firm size and application stock and firm age, respectively) firms tend to produce significantly worse application texts. A good predictive model could thus enable these firms to increase their expected profits significantly while reducing risk for almost no additional cost.

In fact, it is documented that patent success plays a prominent role in venture capital (VC) decision-making, making the process even more important for start-ups in a highly competitive environment (Häussler, Harhoff, and Müller, 2012). Beyond initial funding, patents play an important role in the performance of both VC-backed and non-backed initial public offerings (IPOs) (Cao, Jiang, and Ritter, 2015). Further, Farre-Mensa, Hegde, and Ljungqvist (2020) use an

[11] Topliff v. Topliff. 1892. 145 U.S. 156, Supreme Court of the United States.

IV approach and demonstrate that achieving patent grants causally leads to higher employment and sales growth in startups, and confirm the finding that patents are key to securing funding. Clearly, successful patent grants entail an outsized impact on the survival and performance of small firms and startups in a variety of ways. However, the utility of this application is not limited solely to startups and firms with limited capital. Larger firms often have patents slip through the metaphorical cracks, likely due to the copious number of inventors working simultaneously. For example, some of the "worst" application abstracts in the sample come from innovation heavyweights, as noted in Section 3.2.3.

If the approach of "screening and revision" is feasible, there should be examples in the sample where originally lackluster applications are revised and ultimately accepted with significantly improved patent texts. Thus, I run a test on all application data to search for initially "bad" applications that were turned into "good" patents and accepted. The 500 worst applications from each year in terms of predicted success are gathered and filtered based on ultimate success as patents. The model performs well at identifying the "worst" texts, and only $2,700$ of these roughly $10,000$ applications were accepted.

Of these, I screen for changes in the abstracts. Rather than crude approaches such as checking for literal equality or the number of characters changed, I instead again utilize the ChatGPT textual embeddings, this time to measure change. A major change in the abstract is defined as having a cosine distance of at least 0.05 between the embeddings of the application and the corresponding patent abstract.[12] This provides a sophisticated measure of similarity between texts, capturing changes in meaning, writing style, sentiment, and more - an additional benefit of the textual embedding model. My measure is perhaps more holistic in its evaluation of similarity than previous methods in the literature, such as word vectors employed by Hoberg and Phillips (2016) and word embeddings used in Seegmiller, Papanikolaou, and Schmidt (2023), and while all approaches have their own merit, I choose to leverage the ChatGPT-based measure, given the potential for higher performance – the embedding model takes in the text as a whole rather than as discrete parts. With this added condition of non-trivial changes in the abstract, there is a group of 42 applications throughout the entire sample. I then re-run the prediction model on the textual

[12]This cutoff of 0.05 is a very large change in the text. While cosine distance strictly ranges from 0 to 1, it is very rare and almost impossible for applications to be modified any more and remain the same patent with the same classifications.

embedding of the revised patent grants of these applications.

Panel A of Table 9 reports both predictions of success rate and the "improvement" from application to patent for this sample. I find a mean increase of 10.3% and a median increase of 4.1% in the predicted chance of acceptance, with several applications improving by as much as 50% or more. Both the mean and median results are statistically significant at the 1% level. Even lowering the minimum cosine distance to .02, the documented increase in "quality" is still significant at 1% (Panel B). Also of note is that the average prediction for the revised application subsample is only 17.4%, meaningfully lower than the average of 19.8% in the entire group of $2,700$, indicating that the worst patents tend to be changed the most if they are to be accepted. The results show that revising application texts for a better chance of success is a viable strategy but has likely been underutilized in the past, given the small subsample.

[Insert Table 9 Here]

Figure 5 visualizes the decision-making process of firms applying screening and revision. Currently, firms use the straightforward first method, where inventors write an application and directly submit it to the USPTO after finalizing an invention. I propose an intermediate step in which the firm can repeatedly revise the application text without submitting it to the USPTO using the model until a satisfactory threshold in acceptance chance is reached. Additionally, firms can add a potential second "layer" of screening using the value prediction model. Necessary because the acceptance prediction is not correlated with the value prediction (the USPTO does not accept applications based on estimated economic significance, but rather for originality and non-obviousness), the application of both models could allow firms to maximize both chance of application success and the positive market reaction. I do not find the existence of such trends in the data, likely because existing patent lawyers and experts focus on application acceptance when revising texts. With the introduction of widespread textual analysis and LLM technology, the perception of any publicly available textual information by investors is a pressing concern for all firms, which has been documented in corporate disclosure and the news (Cao, Jiang, Yang, and Zhang, 2023; Huang, Tan, and Wermers, 2020). Particularly, there ought to be a form of "feedback effect" as patent texts are available for public access in bulk data. Thus, the use of machine learning models regarding patent value can help innovating firms monitor the impression they give to institutional investors, pre-emptively adjusting and optimizing for market reaction.

[Insert Figure 5 Here]

A natural question to ask, following these results, is: can AI, in addition to evaluating the quality of applications, help inventors revise their applications? After all, ChatGPT has impressed the world with its ability to perform complex, sophisticated jobs, and has a rich understanding of most subject matters. Moreover, the technology has also demonstrated its capabilities in financial applications in particular (Jha, Qian, Weber, and Yang, 2023; Hansen and Kazinnik, 2023). To answer this question, I again use OpenAI's ChatGPT technology, in this case to revise the worst application abstracts. The "worst" abstracts are defined in this case to be the 50 lowest-rated applications in terms of the embedding-only model prediction from each year of the sample, for a total of 800 applications. The GPT model is prompted with the following text:[13],[14]

> *You are a consultant or lawyer working for a firm wishing to revise a patent application. While the content may not be changed, the presentation and writing style of the application will have a major impact on the chance of acceptance. The application title and abstract will be reviewed for originality, generality, and non-triviality by a neutral patent examiner. Rewrite the following patent application title and abstract, maintaining all important ideas and without adding any additional claims of functionality, so as to maximize chance of acceptance. Major errors are the use of vague language such as the present invention, repetitive sentences and word choice, and over-specification of numbers. You may make any revisions as you see fit, not limited to the above examples. You do not have to keep all specific phrasing or numbers used by the application, but you cannot fabricate new information.*

[Application Title and Abstract]

[13] Particularly, the GPT 3.5-Turbo model available from OpenAI's API at https://platform.openai.com/. This is the full, generative, auto-regressive technology used for the "ChatGPT" service, of which an embedding model is the first component.

[14] Note that it would be statistically unsound to ask ChatGPT to revise the text, then ask ChatGPT to evaluate the revision of the text. However, I use the embedding (which is discussed in Section 3.1) in the deep learning model, which is not connected to ChatGPT, for the creation of *application quality* instead of directly querying ChatGPT. This largely ameliorates the concern, although it would be prohibitively difficult to test a true experiment, i.e., having real patent examiners give blind evaluations of both drafts, for obvious reasons. I am grateful to Jerry Hoberg for identifying this potential issue.

Given the revised text, the same test for *application quality* from Section 3.2 is performed, evaluating the new title and abstract with the neural network trained solely on textual information. Table 10 reports the summary statistics for the initial and post-revision distributions of predicted *application quality* as well as the difference, or improvement. Importantly, both the mean and median improvement are large and significant at the 1% level. Particularly, the mean predicted chance of acceptance increases from 8.0% to 12.8% after revision, a proportional improvement of 60%. Perhaps even more strikingly, for several applications, the *application quality* improves dramatically, exceeding 40 percentage points in multiple cases. While the task of innovation itself cannot, as of yet, be delegated to machine learning, the results certainly suggest that ChatGPT is capable of improving the *writing quality* of existing inventions, increasing the chances of being issued as a grant. Again, this application may be of particular utility to small firms and independent inventors without access to expensive attorneys and consultants.

[Insert Table 10 Here]

4.2. An AI-Adjusted Patent Value

I propose an adjusted measure of the economic value of patents using the predictive model of acceptance. KPSS scale market reaction (filtered to remove noises) by $1/(1-p_0)$, where $p_0 =$ 55% is a constant patent acceptance rate, as the returns measured in a three-day window around acceptance are a reaction only to the roughly 45% chance of denial and not the entire patent. This approach does not account for variation in acceptance rates across significant structural variables such as year, firm size, and industry, as it assumes that the market is unable to predict acceptance. However, institutional investors have long had the capability to read patent applications through brute force application of manpower or even just to use structural variables to predict acceptance. As a result, the market likely already at least partially reflects investors' beliefs about the probability of acceptance, especially because large investors have begun implementing machine learning technology in recent years. In fact, this type of phenomenon (i.e., market anticipation) has been documented in the past by Bhattacharya, Daouk, Jorgenson, and Kehr (2000) and further examined and quantified by Borochin, Celik, Tian, and Whited (2021). Thus, future value estimations of patents ought to consider the predictions and anticipation of the market. For example, if models used by large investors calculate a 95% chance of acceptance for a highly impactful patent, the multiplier is twenty (the value is scaled by 1/(1-.95)). In this example, the previous model assigns the blanket multiplier of 1/(1-.55) and thus substantially underestimates the patent's true value by failing to scale sufficiently. Therefore, I propose a measure of patent value that scales the market reaction by $1/(1 - \hat{p})$, where $\hat{p}$ is the AI model-predicted chance of success of an application.

I test the deviation of the alternative measure from KPSS and report the results in Table 11, compared both to the KPSS assumption of a 55% acceptance rate and an adjusted KPSS measure using a 72.4% acceptance rate, which is the acceptance rate in this paper's final, smaller sample. My scaling factor has a mean proportional difference of 3.65 times and median of 2.46 times from the original KPSS scaling factor; the differences are 2.24 times and 1.51 times, respectively, from the adjusted factor. There is an outsized difference in valuation because of the aforementioned undervaluation by the constant factor in the cases of the "best" patents. To remove outliers, I winsorize my scaling factor and the ultimate valuation at the 1% level. On average, my patent valuation is 22 million dollars larger than the original KPSS value and 17 million larger than the adjusted value. The difference is substantial; for reference, the average KPSS valuation is only 9 million dollars and 14 million when adjusted.

[Insert Table 11 Here]

Next, I conduct comparison tests between the AI-adjusted value and the original KPSS value. If the AI-adjusted value accounts for previously inaccessible information, it ought to provide incremental, additional value as an economic tool for tests. KPSS conduct an important test of the relation between their patent value measure and forward citations (Table 2, KPSS), relating to the literature on patent citations (e.g., Hall, Jaffe, and Trajtenberg, 2005; Abrams, Akcigit, and Grennan, 2019). I perform a similar test with a minor change. While KPSS choose to use all forward citations as the independent variable, I instead choose to implement a 3-year horizon for citation count. This allows for the creation of a clear subsample for the test, specifically, all patents issued prior to 2019, which resolves any truncation concerns (e.g., a patent filed in 2021 will naturally have received fewer citations by 2022 than a similarly impactful patent granted in 2001). All reported test results are qualitatively robust to using citations within other horizons, such as 4 years and 10 years. Any longer would excessively restrict the regression sample, while shorter horizons do not carry sufficient information, given the time it takes for applications written after the issuance of a certain patent to be granted themselves.

I first estimate the following regression for firm-application-time (i, j, t):

$$\mathit{Value\ Metric}_{i,j,t} = \beta_1 \log(1 + C_{j,t}) + \beta_2 \mathit{Size}_{i,t} + \gamma \cdot Controls_{j,t} + \delta_i + \delta_t + \epsilon_{i,j,t}. \tag{5}$$

Table 12 Panel A reports the results. In Columns (1) and (2), the value metrics tested are the KPSS measure and the AI-adjusted value measure. Results show that both are significantly and positively correlated with forward citations. Given that the difference between the AI-adjusted value and KPSS value arises from the scaling factor $1/(1 - \hat{p})$, where $\hat{p}$ is the AI-predicted success rate of the original patent application, I also estimate regression (5) with the AI scaling factor and the predicted chance $\hat{p}$ itself, as the dependent variable. Columns (3) and (4) show that both the scaling factor and the predicted success rates are significantly and positively related to patent citations with higher t-statistics (4.78, 5.84) than that of KPSS value (2.71). Panel B of Table 12 shows that although the AI-adjusted value has a high positive correlation (0.503) with the KPSS value, both the AI scaling factor and the predicted success rate have a low and negative correlation with the KPSS value (-0.036 and -0.092, respectively).

[Insert Table 12 Here]

To further disentangle the effects of the AI-adjusted value and the KPSS value, I regress patent citations simultaneously on both the AI scaling factor/predicted chance of success and the KPSS value.

$$\begin{aligned} \log(1 + C_{j,t}) = \beta_1 AI\ Variable_{j,t} + \beta_2 KPSS\ Value_{i,j,t} + \beta_3 \mathit{Size}_{i,t} \\ + \gamma \cdot Controls_{j,t} + \delta_i + \delta_t + \epsilon_{i,j,t}. \end{aligned} \tag{6}$$

The $AI\ Variable$ above is either the AI scaling factor or the AI-predicted chance of success (I do not include the AI-predicted value and the KPSS in the same regression due to their high correlation). This test helps identify if the AI-predicted value is providing additional information to that which is already found in the KPSS measure. The findings, reported in Panel C of Table 12, suggest that this is the case. In both cases, the association between citations and the AI-predicted variable is statistically significant, with a higher significance than that of KPSS.

Overall, the evidence suggests the following. First, rather than recreating the KPSS measure through an alternative channel, the AI scaler provides a measure of the patent from an orthogonal perspective, capturing distinct factors and trends. This suggests that the AI-adjusted value combines two economically significant, distinct measures of patent quality and importance, providing incremental information to the KPSS value. Second, my findings are in line with previous hypotheses in the literature that there may be distinct components to a patent's value. Particularly, the characteristics that the USPTO prioritizes in its evaluation of applications may differ significantly from those emphasized by investors and, therefore, by proxy, the firm itself.

An important fact to note is that, while there is strong, significant association between the AI-based metrics and citations when regressing citations on the metrics, it is not feasible to use the AI-based metrics to directly predict citations as the R-squared in the regressions in Panel C but without firm and year fixed effects (i.e., restricting to ex-ante available information) is around 2% (untabulated). This answers the question: if the AI-based value metrics from the *application* model are highly correlated with citations, what is the purpose of creating a distinct model for *citations* in Section 3.4? Because of the weak R-squared value, these factors alone can not function as a replacement for the machine learning techniques I use to predict citations.

4.3. Patent Application Strength and Firm Performance

The ability of firms to file successful patent applications is critical and ex-ante unobservable. Denied applications can theoretically be of extreme negative value to firms, as they lead to failure in commercialization and, more significantly, allow competitors to "steal" the idea. Pre-emptive knowledge of successful innovation through the "best" applications is also highly valuable. However, there is, to the best of my knowledge, no measure of the quality and strength of firms' current patent applications, as the literature has focused primarily on the value of patent grants.

I define a firm-level *application strength* measure as the mean predicted chance of acceptance for all patent applications published by a firm in a given month, multiplied by the square root of the number of applications in that month.[15] This measure takes into account both the quality and quantity of firms' innovation, as both are important components of innovative productivity. The endogenous growth theory (Klette and Kortum, 2004; Acemoglu et al., 2018) predicts that firms' innovation drives their productivity, growth, and returns. Given the novel, ex-ante nature of the

[15] I elect to use square root because the number of applications has a skewed distribution with heavy tails. The results are robust to using the logarithm of the number of applications instead of its square root as weights in the measure.

application strength measure, I expect it to be positively related to firms' future performance.

I thus construct a long-short portfolio based on the measure. Each month, firms are sorted into two groups based on the median of *application strength*. The portfolio holds stocks in the above-median group and shorts stocks in the below-median group for a one-month horizon and is rebalanced monthly. The abnormal returns of the portfolio are calculated based on the Fama-French three-, four-, and five-factor models (Fama and French, 1993, 2015; Carhart, 1997).

Table 13 reports the alphas of the portfolio. The abnormal returns of the long and short portfolios are independently significant at the 10% level. The difference in Fama-French four-factor adjusted returns, or overall performance of the long-short portfolio, is 3.3% annually, statistically significant at the 5% level. The returns adjusted by the three- and five-factor models are also significant and similar in magnitude.

Overall, the evidence suggests that the application strength measure provides important ex-ante information about the latent innovative capability of firms. The results are largely consistent with findings in the literature that the market underreacts to early-stage R&D relative to the true value of latent innovation, i.e., innovation and research which has not yet been commercialized or, in this case, granted as a patent (Eberhart, Maxwell, and Siddique, 2004). Another complementary channel is the short term-ism consistently exhibited by managers at investment funds. Given that patent applications often take years to realize as successful innovation outcomes, career concerns of "myopic" managers may lead to systematic mispricing (underpricing) of these long-term factors (Shleifer and Vishny, 1990; Chevalier and Ellison, 1999; Agarwal, Vashishtha, and Venkatachalam, 2018).

[Insert Table 13]

5. Concluding Remarks

This paper adopts a new approach to patent analysis with the groundbreaking ChatGPT technology, moving beyond conventional structural variables. OpenAI's state-of-the-art textual embedding allows deep learning models to interpret previously inaccessible information about the impact of each distinct invention. I show a 24% incremental improvement in R-squared predicting patent value when adding the embedding vector, and 12% when predicting patent citations. The full model for predicting acceptance is able to clearly isolate the worst and best applications with a roughly 10% improvement in both over the benchmark, which is critical in economic contexts. I propose a revision of the widely accepted Kogan et al. (2017) valuation of patents. My measure has a mean deviation of 22 million dollars, or 2.5 times from the KPSS measure and accounts for institutional analysis of patents by providing an alternative to the hypothesis of uniform acceptance rate. Additionally, the models provide an opportunity to firms, especially startups or small firms, to enhance their patenting and innovation processes. Finally, a long-short portfolio constructed based on the strength of applications filed by firms achieves significant annual returns of 3.3%, indicating that the application strength measure contains important information about firms' innovative productivity and future growth.

This paper also leaves room in the future for further research alongside the development of machine learning and natural language processing technologies. First, LLMs currently need to be better equipped for even basic symbolic processing (as can be seen when asking ChatGPT to perform any mathematical calculations) and are definitively unqualified for evaluation of actual engineering or design quality. In the case of patent classifications, the ability for symbolic processing could enable a model to incorporate richer information about the invention. Second, GPT-4, along with other "general" developments such as Gato, are the first step towards multimodal (accepting non-textual, and, specifically, image input) models, which will be able to leverage the remaining components of a patent, i.e., figures and drawings, which are, in theory, potentially as important as the text. Finally, there are many unexplored further applications of many of the techniques I develop, from ChatGPT text-revision writ large to my measure for patent value.

References

Abrams, D., U. Akcigit, and J. Grennan. 2019. Patent value and citations: Creative destruction or strategic disruption? Working Paper, Pennsylvania Wharton, Chicago Economics, Berkeley Haas.

Acemoglu, D., U. Akcigit, H. Alp, N. Bloom, and W. Kerr. 2018. Innovation, reallocation, and growth. *American Economic Review* 108:3450–91.

Acikalin, U. U., T. Caskurlu, G. Hoberg, and G. M. Phillips. 2022. Intellectual property protection lost and competition: An examination using machine learning. Working Paper, TOBB Economics, Amsterdam Business, USC Marshall, Dartmouth Tuck.

Agarwal, V., R. Vashishtha, and M. Venkatachalam. 2018. Mutual fund transparency and corporate myopia. *Review of Financial Studies* 31:1966–2003.

Austin, D. H. 1993. An event-study approach to measuring innovative output: The case of biotechnology. *American Economic Review* 83:253–8.

Bhagat, S., and P. A. Frost. 1986. Issuing costs to existing shareholders in competitive and negotiated underwritten public utility equity offerings. *Journal of Financial Economics* 15:233–59.

Bhattacharya, U., H. Daouk, B. Jorgenson, and C.-H. Kehr. 2000. When an event is not an event: The curious case of an emerging market. *Journal of Financial Economics* 55:69–101.

Biasi, B., and S. Ma. 2022. The education-innovation gap. Working Paper, Yale Management.

Bloom, N., M. Schankerman, and J. Van Reenen. 2013. Identifying technology spillovers and product market rivalry. *Econometrica* 81:1347–93.

Borochin, P., M. A. Celik, X. Tian, and T. M. Whited. 2021. Identifying the heterogeneous impact of highly anticipated events: Evidence from the tax cuts and jobs act. Working Paper, University of Florida Finance, University of Toronto Economics, UGA Terry, Michigan Ross.

Bowen, D. E., L. Frésard, and G. Hoberg. 2023. Rapidly evolving technologies and startup exits. *Management Science* 69:940–67.

Box, G. E., and D. R. Cox. 1964. An analysis of transformations. *Journal of the Royal Statistical Society Series B: Statistical Methodology* 26:211–43.

Brown, T., B. Mann, N. Ryder, et al. 2020. Language models are few-shot learners. *Advances in Neural Information Processing Systems* 33:1877–901.

Bybee, L., B. T. Kelly, A. Manela, and D. Xiu. 2023. Business news and business cycles. *Journal of Finance* forthcoming.

Cao, J., F. Jiang, and J. R. Ritter. 2015. Patents, innovation, and performance of venture capital-backed ipos. Working Paper, CUFE, University of Florida Finance.

Cao, S., W. Jiang, J. L. Wang, and B. Yang. 2022. From man vs. machine to man + machine: The art and AI of stock analyses. Working Paper, Maryland Smith, Emory Goizueta, LSU, GSU Robinson.

Cao, S., W. Jiang, B. Yang, and A. L. Zhang. 2023. How to talk when a machine is listening?: Corporate disclosure in the age of AI. *Review of Financial Studies* forthcoming.

Carhart, M. M. 1997. On persistence in mutual fund performance. *Journal of Finance* 52:57–82.

Chen, L., M. Pelger, and J. Zhu. 2023. Deep learning in asset pricing. *Management Science* forthcoming.

Chevalier, J., and G. Ellison. 1999. Career concerns of mutual fund managers. *Quarterly Journal of Economics* 114:389–432.

Cohen, L., K. Diether, and C. Malloy. 2013. Misvaluing innovation. *Review of Financial Studies* 26:635–66.

Cong, L. W., G. Feng, J. He, and X. He. 2023. Asset pricing with panel tree under global split criteria. Working Paper, Cornell Johnson, CityU.

Cong, L. W., T. Liang, and X. Zhang. 2020. Textual factors: A scalable, interpretable, and data-driven approach to analyzing unstructured information. Working Paper, Cornell Johnson, Chicago Booth.

Cong, L. W., K. Tang, J. Wang, and Y. Zhang. 2021. Alphaportfolio: Direct construction through reinforcement learning and interpretable AI. Working Paper, Cornell Johnson, Tsinghua, Beihang.

Costello, A. M., A. K. Down, and M. N. Mehta. 2020. Machine + man: A field experiment on the role of discretion in augmenting AI-based lending models, *Journal of Accounting and Economics* 70:101360.

Drucker, P. 2014. *Innovation and entrepreneurship*. Routledge.

Eberhart, A. C., W. F. Maxwell, and A. R. Siddique. 2004. An examination of long-term abnormal stock returns and operating performance following R&D increases. *Journal of Finance* 59:623–50.

Fama, E. F., and K. R. French. 1993. Common risk factors in the returns on stocks and bonds. *Journal of Financial Economics* 33:3–56.

———. 2015. A five-factor asset pricing model. *Journal of Financial Economics* 116:1–22.

Farre-Mensa, J., D. Hegde, and A. Ljungqvist. 2020. What is a patent worth? Evidence from the US patent “lottery”. *Journal of Finance* 75:639–82.

Fitzgerald, T., B. Balsmeier, L. Fleming, and G. Manso. 2021. Innovation search strategy and predictable returns. *Management Science* 67:1109–37.

Freyberger, J., A. Neuhierl, and M. Weber. 2020. Dissecting characteristics nonparametrically. *Review of Financial Studies* 33:2326–77.

Gu, S., B. Kelly, and D. Xiu. 2020. Empirical asset pricing via machine learning. *Review of Financial Studies* 33:2223–73.

Hall, B. H., A. Jaffe, and M. Trajtenberg. 2005. Market value and patent citations. *RAND Journal of Economics* 16–38.

Hanley, K. W., and G. Hoberg. 2019. Dynamic interpretation of emerging risks in the financial sector. *Review of Financial Studies* 32:4543–603.

Hansen, A. L., and S. Kazinnik. 2023. Can ChatGPT decipher fedspeak? Working Paper, Federal Reserve Board.

Häussler, C., D. Harhoff, and E. Müller. 2012. To be financed or not. . . –the role of patents for venture capital-financing. Working Paper, University of Passau, CEPR, Lille IESEG Management.

Hirshleifer, D., P.-H. Hsu, and D. Li. 2013. Innovative efficiency and stock returns. *Journal of Financial Economics* 107:632–54.

Hoberg, G., and G. Phillips. 2016. Text-based network industries and endogenous product differentiation. *Journal of Political Economy* 124:1423–65.

Huang, A. G., H. Tan, and R. Wermers. 2020. Institutional trading around corporate news: Evidence from textual analysis. *Review of Financial Studies* 33:4627–75.

Huang, A. H., H. Wang, and Y. Yang. 2023. FinBERT: A large language model for extracting information from financial text. *Contemporary Accounting Research* 40:806–41.

Jha, M., J. Qian, M. Weber, and B. Yang. 2023. ChatGPT and corporate policies. Working Paper, GSU Robinson, Chicago Booth.

Kelly, B., D. Papanikolaou, A. Seru, and M. Taddy. 2021. Measuring technological innovation over the long run. *American Economic Review: Insights* 3:303–20.

Kim, A. G., M. Muhn, and V. V. Nikolaev. 2023. Bloated disclosures: Can ChatGPT help investors process information? Working Paper, Chicago Booth.

Klette, T. J., and S. Kortum. 2004. Innovating firms and aggregate innovation. *Journal of Political Economy* 112:986–1018.

Kline, P., N. Petkova, H. Williams, and O. Zidar. 2019. Who profits from patents? Rent-sharing at innovative firms. *Quarterly Journal of Economics* 134:1343–404.

Kogan, L., D. Papanikolaou, A. Seru, and N. Stoffman. 2017. Technological innovation, resource allocation, and growth. *Quarterly Journal of Economics* 132:665–712.

Lanjouw, J. O., and M. Schankerman. 2001. Characteristics of patent litigation: A window on competition. *RAND Journal of Economics* 129–51.

Li, K., F. Mai, R. Shen, and X. Yan. 2021. Measuring corporate culture using machine learning. *Review of Financial Studies* 34:3265–315.

Mezzanotti, F., and T. Simcoe. 2023. Research and/or development? Financial frictions and innovation investment. Working Paper, Northwestern Kellog, BU Questrom.

Misra, D. 2019. Mish: A self regularized non-monotonic activation function. Working Paper, arXiv.

Pakes, A. 1985. On patents, R&D, and the stock market rate of return. *Journal of Political Economy* 93:390–409.

Radford, A., K. Narasimhan, T. Salimans, et al. 2018. Improving language understanding by generative pre-training. Working Paper, OpenAI.

Ramachandran, P., B. Zoph, and Q. V. Le. 2017. Searching for activation functions. Working Paper, Google AI.

Seegmiller, B., D. Papanikolaou, and L. D. Schmidt. 2023. Measuring document similarity with weighted averages of word embeddings. *Explorations in Economic History* 87.

Shleifer, A., and R. W. Vishny. 1990. Equilibrium short horizons of investors and firms. *American Economic Review* 80:148–53.

Xiao, L., G. Wang, and Y. Zuo. 2018. Research on patent text classification based on word2vec and LSTM. *IEEE* 2018 11th International Symposium on Computational Intelligence and Design (ISCID).

Zheng, X. 2022. How can innovation screening be improved? A machine learning analysis with economic consequences for firm performance. Working Paper, University of Connecticut Finance.

Appendix A. Deep Learning Model Design and Implementation

When implementing my neural network model, I create three intermediate layers, setting node counts for each layer as a roughly geometric series (the final layer has no activation, and can be thought of as an OLS regression on the dependent from the penultimate layer). For the models with the large embedding, this manifests as a 200-30-3-1 layer design. For the benchmark models trained only on structural variables, the layers are instead set at 10-3-1, because of the significantly lower feature count. I utilize 20% dropout regularization after the first layer on all models to prevent overfitting (the curse of dimensionality, leading to near interpolation of the training sample and poor performance out-of-sample), a process which essentially "drops" 20% of the nodes every epoch of training by setting their parameters to 0. This outperformed other methods of regularization such as L1 and L2, as well as significantly reducing node count and removing regularization. This is likely because it better maintains high-degree interactions within the embedding and avoids the model becoming over-fixated with the individually more impactful structural variables, as could occur with L1 and L2. For the value and citation predictions, MSE is minimized, while the success prediction uses binary cross-entropy as the target. As the success prediction model outputs a real number in the range $(0, 1)$, all predictions above 50% (below 50%) are treated as success predictions, or 1 (failure predictions, or 0). Models are trained over 100 epochs with a batch size of 500. The Adam optimizer is used, a standard optimizer that parameterizes the learning rate to obtain the best possible results.

The gradient boosting benchmark is trained using the "xgboost" python package. The hyper-parameters are as follows: subsample = 1, colsample by tree = 0.1, minimum child weight = 15, max depth = 5, learning rate = .01, number of estimators = 100, early stopping rounds = 10, which produce the best results out of multiple tested configurations.

Appendix B. Example Applications with Low Predicted Chance of Success

Exhibit 1.

Title: Pharmaceutical compositions and methods comprising combinations of 2-alkylidene-19-nor-vitamin D derivatives and parathyroid hormone

Abstract: The present invention relates to pharmaceutical compositions and methods of treatment comprising administering to a patient in need thereof a combination of a 2-alkylidene-19-nor-vitamin D derivative and parathyroid hormone or an active fragment or variant thereof. Particularly, the present invention relates to pharmaceutical compositions and methods of treatment comprising administering to a patient in, need thereof 2-methylene-19-nor-20(S)-1alpha,25-dihydroxyvitamin D3 and parathyroid hormone or an active fragment or variant thereof.

Exhibit 2.

Title: Method 741

Abstract: Methods of treatment and pharmaceutical compositions for providing improved cognition in subjects suffering from schizophrenia, Alzheimer's disease or other conditions with impaired cognitive function.

Exhibit 3.

Title: Treatments Using Venlafaxine

Abstract: This invention provides a method of treating obesity, generalized anxiety disorder, post-traumatic stress disorder, late luteal phase dysphoric disorder (premenstrual, syndrome), attention deficit disorder, with and without hyperactivity, Gilles de la Tourette syndrome, bulimia nervosa or Shy Drager Syndrome in a: mammal by administering to the mammal an effective amount of a hydroxycycloalkanephenethyl amine of the following structural formula:

[Figure Omitted]

in which A is a moiety of the formula

[Figure Omitted]

where the dotted line represents optional unsaturation; R1 is hydrogen or alkyl; R2 is alkyl; R4 is hydrogen, alkyl, formyl, or alkanol; R5 and R6 are, independently, hydrogen, hydroxyl, alkyl, alkoxy, alkanoyloxy, cyano, nitro, alkylmercapto, amino, alkylamino, dialkylamino, alkanamido, halo, trifluoromethyl, or taken together, methylene dioxy; R7 is hydrogen or alkyl; and n is 0, 1, 2, 3, or 4; or a pharmaceutically acceptable salt thereof.

Exhibit 4.

Title: Combination of Laquinimod and Pridopidine for Treating Neurodegenerative Disorders, in Particular Huntington's Disease

Abstract: This invention provides a method of treating a patient afflicted with a neurodegenerative disorder, e.g., Huntington's disease (HD), comprising administering to the patient laquinimod as an add-on therapy to or in combination with pridopidine. This invention also provides a package and a pharmaceutical composition comprising laquinimod and pridopidine for treating a patient afflicted with a neurodegenerative disorder, e.g., HD. This invention also provides laquinimod for use as an add-on therapy or in combination with pridopidine in treating a patient afflicted with a neurodegenerative disorder, e.g., HD. This invention further provides use of laquinimod and pridopidine in the preparation of a combination for treating a patient afflicted with a neurodegenerative disorder, e.g., HD.

Exhibit 5.

Title: Predicting Customer Satisfaction

Abstract: Systems and methods for predicting customer satisfaction are disclosed. An example method includes identifying business factors related to customer satisfaction. The method also includes translating the business factors to measurable metrics. The method also includes predicting for a user, variations in performance leading to lower customer satisfaction, based on the measurable metrics.

Figure 1. Time-Series Variation in Aggregate Acceptance Rates of Patent Applications

This figure documents the variation in average patent application acceptance rates across all years of the sample. Variation in colors corresponds to variation in magnitude, with red as low and blue as high.

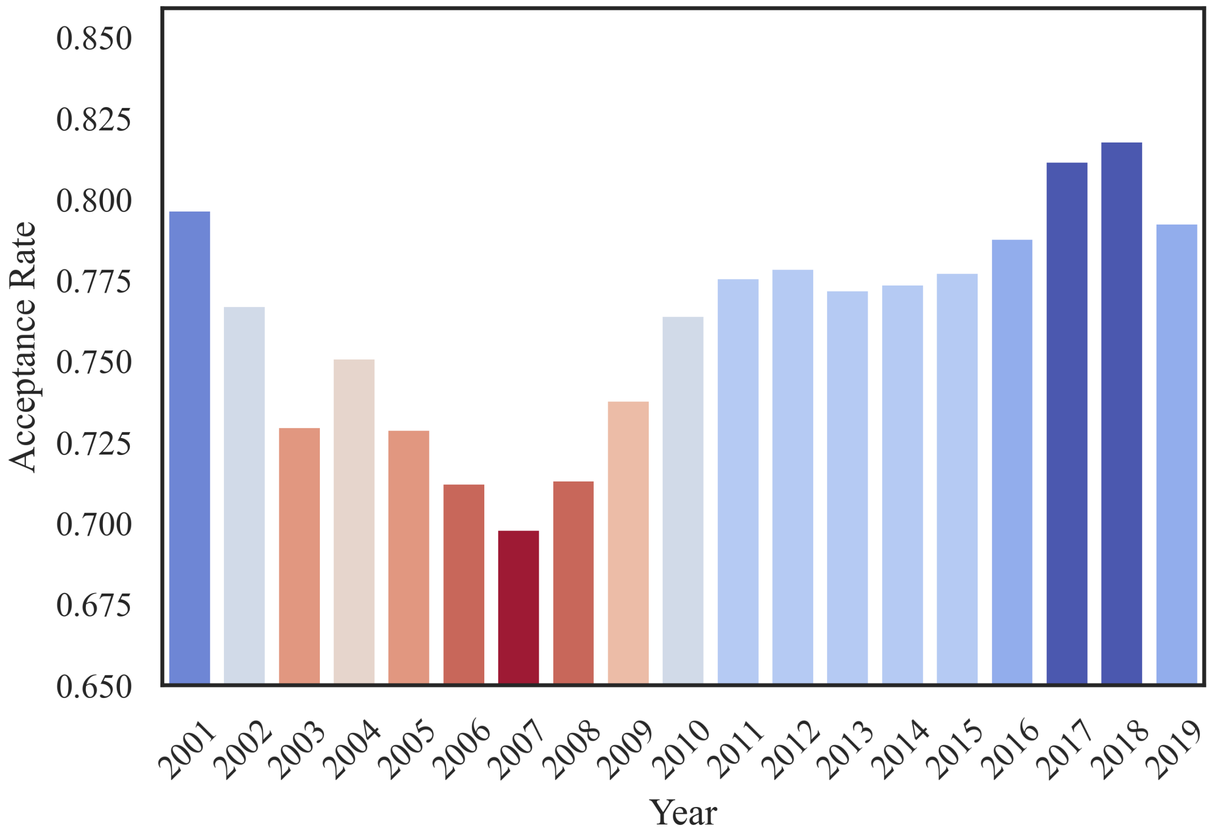


Figure 2. Comparison of Activation Functions

This figure plots several common activation functions, *ELU* and *ReLU*, along with the contemporary *Swish* function in comparison to *Mish*. Simple *ReLU* is the most used activation function, which equals 0 if x is negative, and x otherwise. *ELU* is defined as $e^{x}\breve{}1$ for negative x, and x for positive x, while *Swish* is defined as $x \cdot sigmoid(\beta x)$, where β is learned. *Mish* is defined as $x \cdot \tanh(\ln(1 + e^{x}))$.

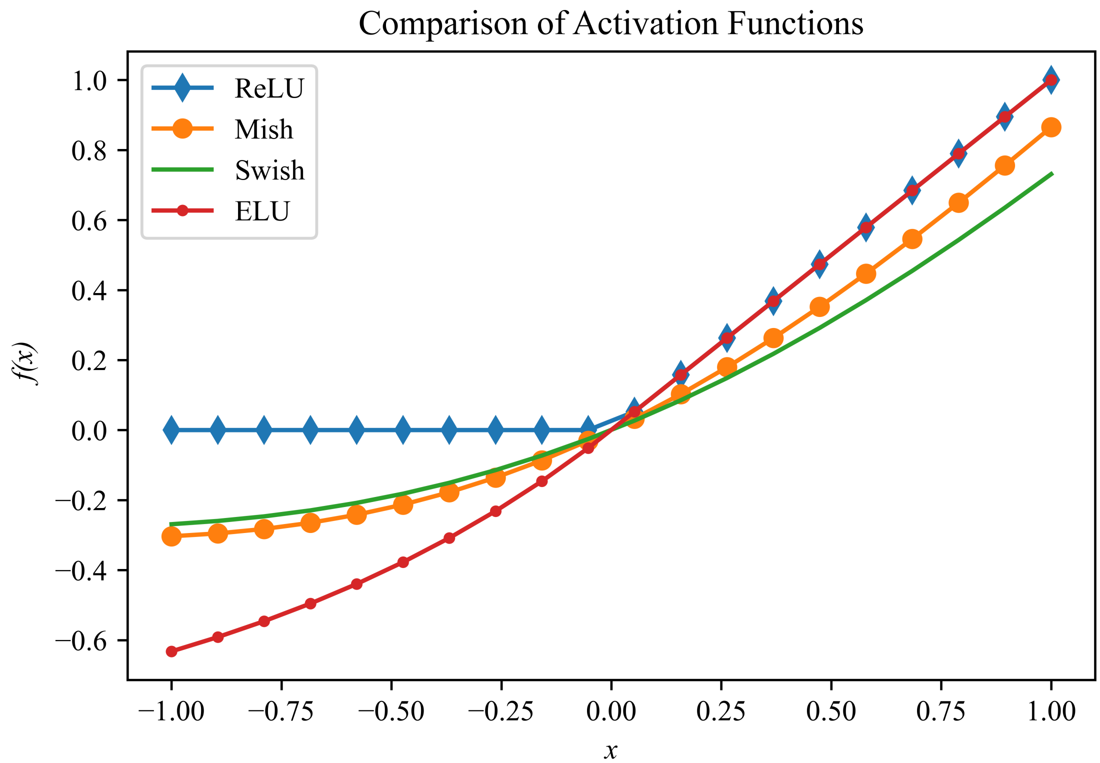
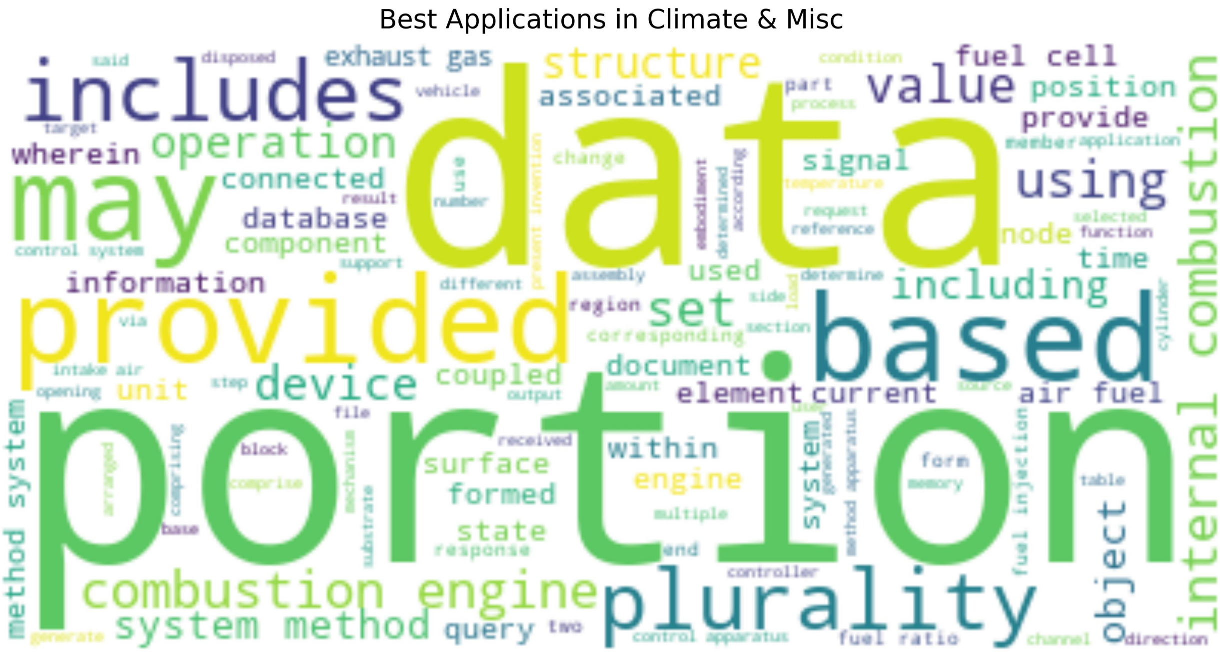


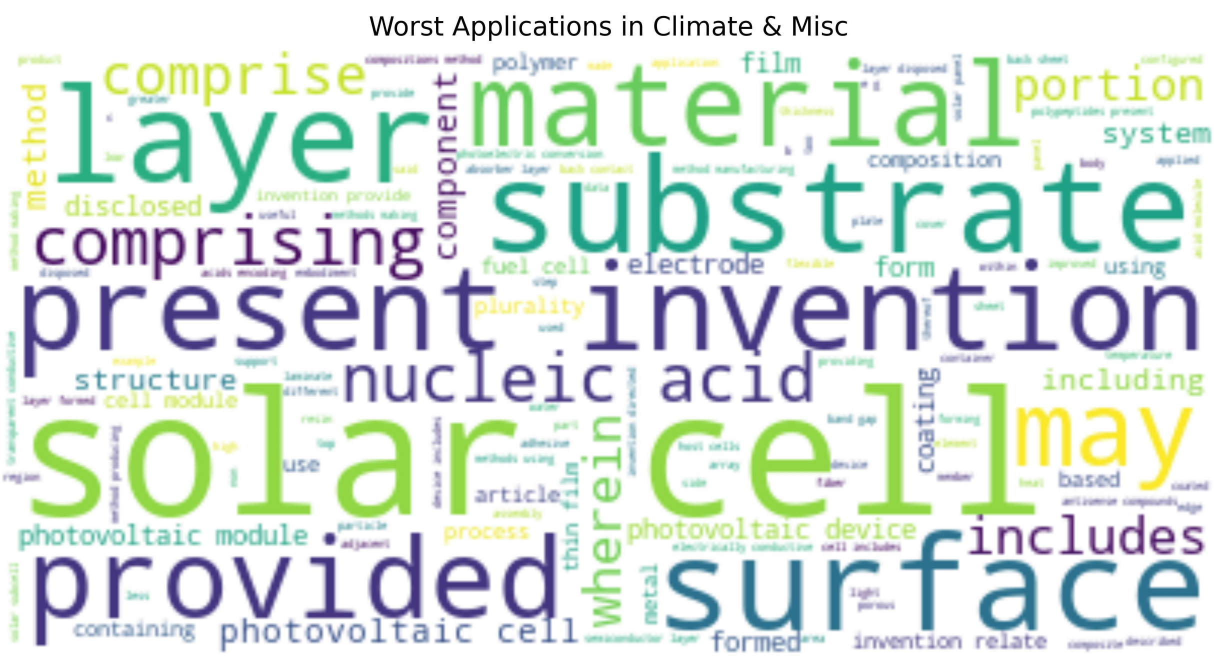
Figure 3. Word Clouds of Best and Worst Applications

This figure documents the "word clouds" for the title and abstracts of the 30,000 worst and best applications (defined as having, respectively, the lowest and highest predicted chance of success by the full model) for applications in the Climate, Biotech, and ICT categories.

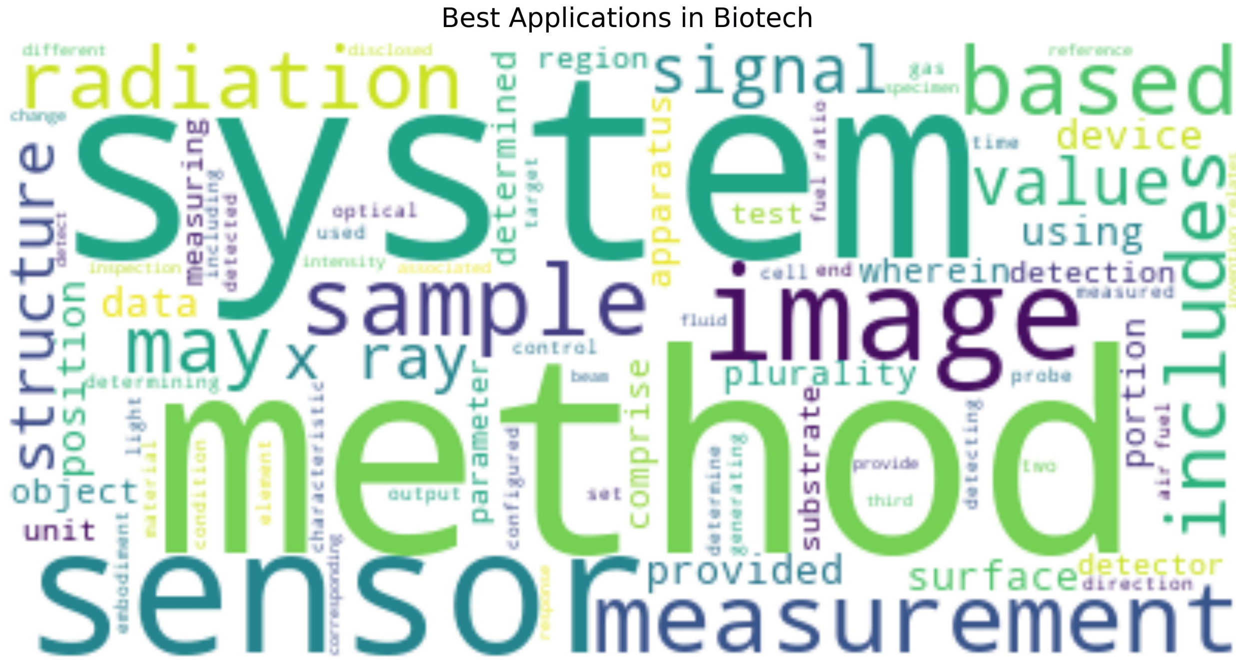
Panel 1A: Best in Climate and Misc.



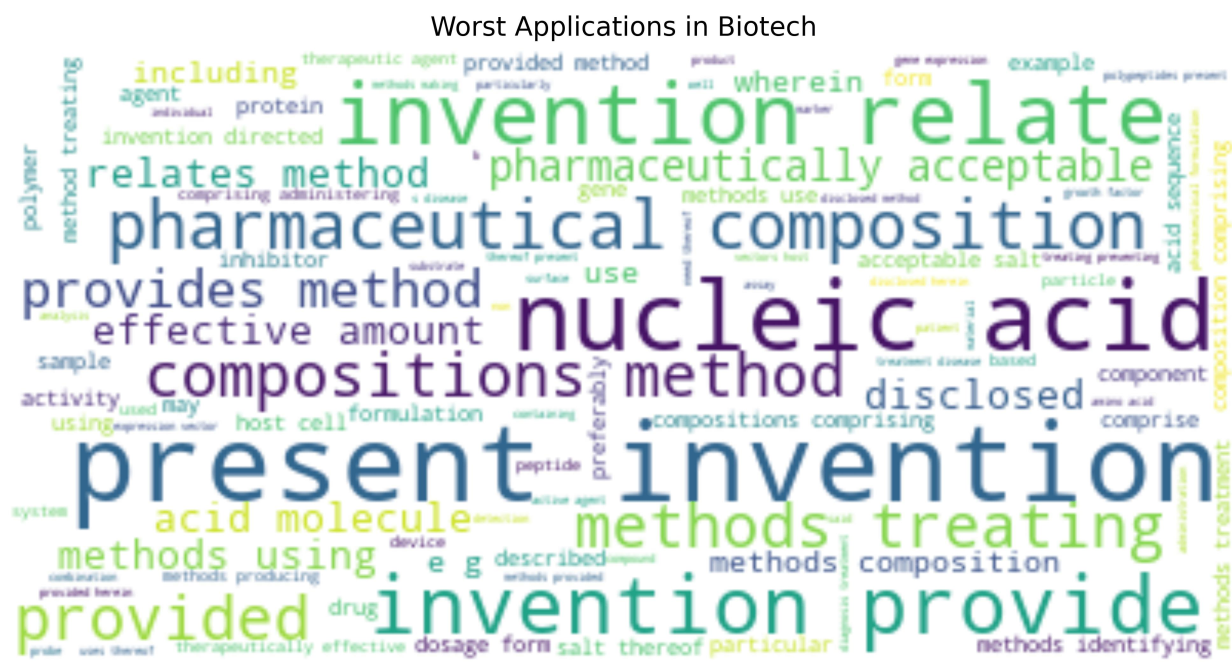
Panel 1B: Worst in Climate and Misc.



Panel 2A: Best in Biotech.



Panel 2B: Worst in Biotech.



Panel 3A: Best in ICT.

Panel 3B: Worst in ICT.

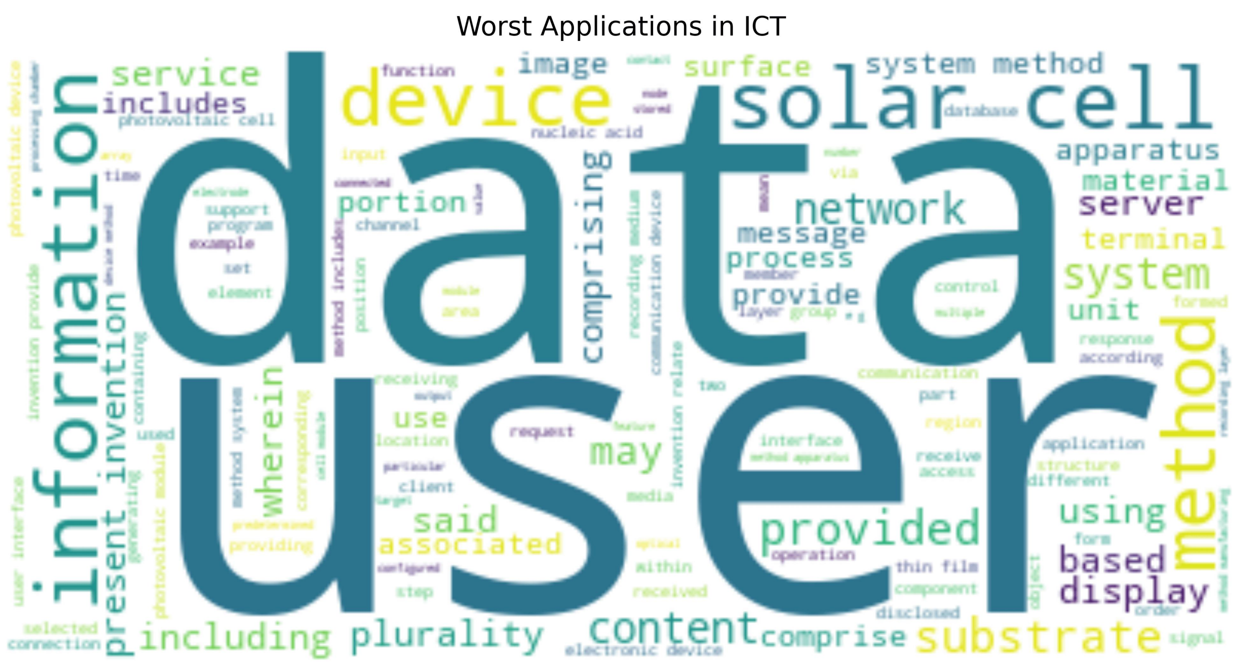
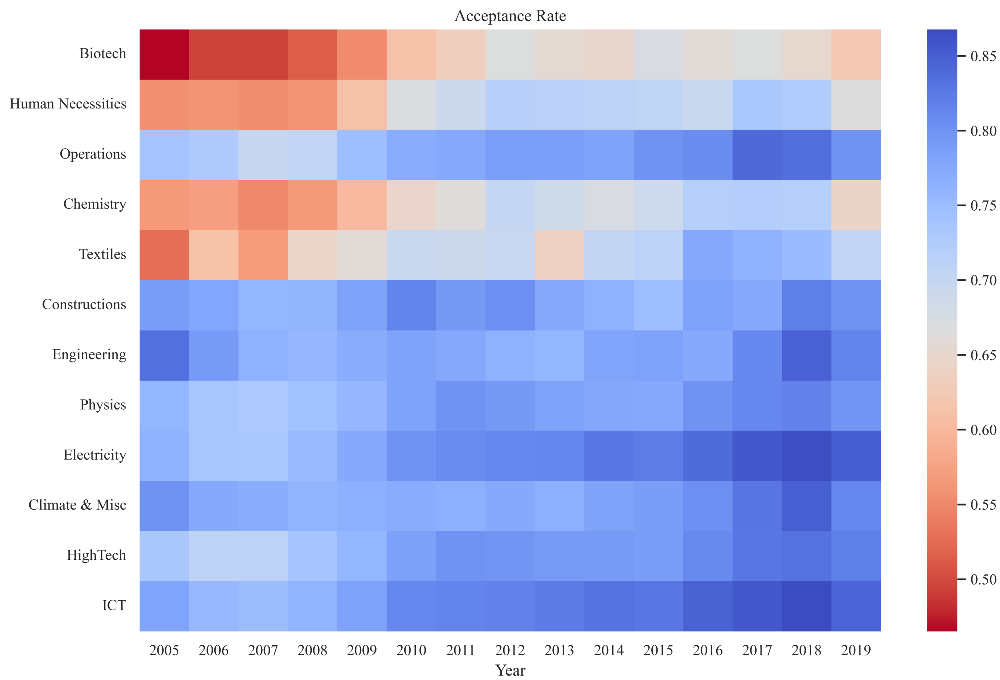


Figure 4. Acceptance Rates and Application Quality by Patent Classification and Time

This figure documents the variation in acceptance rates and application quality across different patent classifications (CPC and generated) and years. *Application quality* is the predicted chance of acceptance from a neural network trained on only textual information, which isolates the quality of the text in particular.

Panel A: Acceptance Rate.



Panel B: Application Quality.

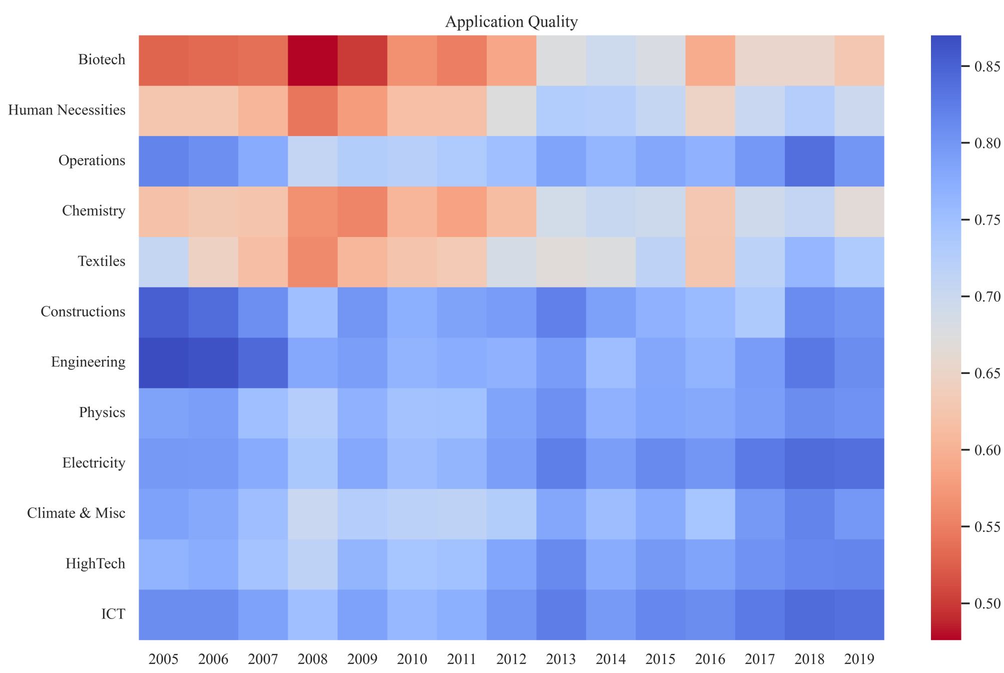


Figure 5. Potential Process of Application Screening

This figure displays a potential decision-making process for firms using AI models for "application screening." The first line shows the status quo, while the second option involves using a predictive model for success to "screen" applications. The third line adds a second layer of screening using a value predictive model, which is necessary as value is not directly correlated with acceptance.

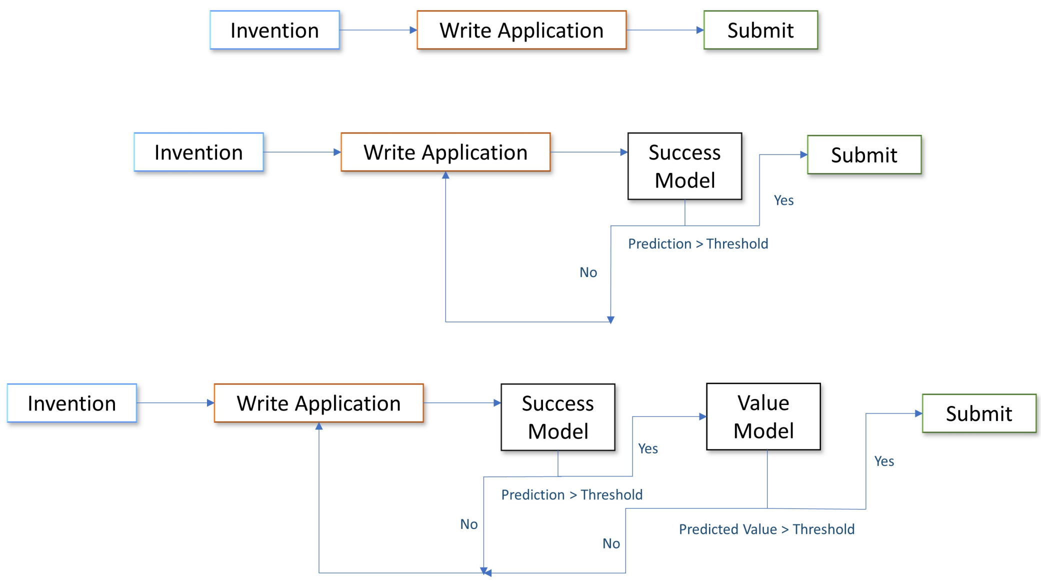


Table 1. Application Success Prediction Results

This table shows the year-by-year performance of the primary predictive model for patent application success, using the machine learning model and the full list of features, including the embedding of the application text. AUC is the area under the Receiver Operating Characteristic (ROC) curve and measures the probability of ranking a positive instance higher than a negative instance. Accuracy is the overall percentage of correct predictions. Precision is the percentage of true positives in all labeled positives, and Recall is the percentage of true positives in all real positives. F1 Score is the harmonic mean of Precision and Recall. Mean and median are calculated across all years.

Year	AUC	F1 Score	Accuracy	Precision	Recall
2004	60.9%	85.9%	77.0%	79.7%	93.1%
2005	57.9%	84.8%	75.1%	76.3%	95.3%
2006	58.8%	82.8%	72.8%	75.4%	91.7%
2007	58.9%	81.9%	71.8%	74.2%	91.4%
2008	57.6%	82.8%	72.5%	74.9%	92.5%
2009	57.2%	84.2%	74.3%	76.9%	93.1%
2010	58.6%	84.9%	75.4%	80.0%	90.4%
2011	55.6%	86.9%	77.6%	79.7%	95.6%
2012	55.5%	87.4%	78.4%	79.9%	96.6%
2013	56.6%	86.9%	77.7%	79.7%	95.4%
2014	56.4%	87.3%	78.4%	79.8%	96.5%
2015	56.9%	87.6%	78.8%	80.3%	96.5%
2016	55.9%	88.2%	79.6%	80.9%	97.0%
2017	56.2%	89.4%	81.4%	83.2%	96.6%
2018	53.8%	89.9%	81.9%	83.0%	98.0%
2019	55.0%	88.0%	79.2%	81.0%	96.3%
Mean	57.0%	86.2%	77.0%	79.1%	94.7%
Median	56.8%	86.9%	77.7%	79.8%	95.5%

Table 2. Application Success Benchmark Comparison Results

This table compares the mean and median performance of the full model evaluated in Table 1, a gradient boosting classifier trained on the full list of features, and a neural network trained with all features except the embedding of the application text. The performance metrics are defined in Table 1.

		AUC	F1 Score	Accuracy	Precision	Recall
Full Model	Mean	57.0%	86.2%	77.0%	79.1%	94.7%
	Median	56.8%	86.9%	77.7%	79.8%	95.5%
XGBoost	Mean	51.4%	86.5%	76.5%	76.8%	99.0%
	Median	50.3%	87.2%	77.3%	77.3%	99.8%
No Embed	Mean	52.6%	86.4%	76.6%	77.2%	98.2%
	Median	52.2%	87.2%	77.4%	77.7%	98.8%

Table 3. Comparison of Best and Worst Predicted Applications

This table demonstrates the difference in performance between the full model and the benchmark trained without the embedding features when attempting to identify the "best" and "worst" applications. I present the average acceptance rate across all years of the full model and benchmark for the yearly top and bottom 100, 250, 500, and 1000 applications. The top and bottom applications are defined as having the highest and lowest predicted chance of acceptance by the two models.

	Best Success			Worst Success		
Yearly Cutoff	Full Model	No Embedding	Difference	Full Model	No Embedding	Difference
100	96.1%	87.0%	9.0%	26.4%	37.1%	-10.8%
250	95.9%	86.9%	9.0%	27.1%	38.0%	-11.0%
500	95.1%	86.6%	8.5%	29.7%	40.0%	-10.3%
1000	94.1%	86.7%	7.4%	34.5%	43.5%	-9.0%

Table 4. Application Quality and Firm Characteristics

This table reports the results of the regressions of *application quality* on corporate variables. *Application quality* is proxied through the predicted chance of acceptance by a model trained only on the ChatGPT embedding vector (so that the model only has the textual information as inputs). Firm age is defined as years since the firm was first listed on CRSP. Size is the natural logarithm of shares outstanding times price, adjusted for inflation. Application stock is defined as the cumulative number of all patent applications filed prior to the month of the current application, regardless of success. Patent controls include classifications (CPC and generated) and the natural logarithm of the number of classifications an application belongs to. T-statistics are reported in parentheses, with standard error clustered at the firm level. ***, **, * denote statistical significance at the 0.01, 0.05, and 0.10 levels, respectively.

	Application Quality				
	(1)	(2)	(3)	(4)	(5)
Firm Size	0.016***			0.013***	0.012***
	(5.37)			(4.72)	(4.90)
Firm Age		0.056***		0.025**	0.028***
		(6.73)		(2.53)	(3.04)
Application Stock			0.016***	0.011***	0.011***
			(6.02)	(3.04)	(3.18)
Constant	0.522***	0.571***	0.637***	0.400***	0.407***
	(11.65)	(20.00)	(30.39)	(9.33)	(9.72)
Patent Controls	-	-	-	-	Y
Firm Fixed Effects	Y	Y	Y	Y	Y
N	722,163	722,163	722,163	722,163	722,163
Adj. R^2	0.265	0.268	0.268	0.271	0.296

Table 5. Primary Value Prediction Results

This table reports the year-by-year results of my prediction model for KPSS valuation, using machine learning and the full list of features, including the embedding of the application text. I report R-squared score, adjusted R-squared, and mean squared error, as well as overall mean and median for all statistics.

Year	R^2	Adj. R^2	MSE
2004	38.4%	36.9%	20.50
2005	40.7%	39.0%	22.41
2006	41.0%	39.6%	17.89
2007	44.0%	42.4%	16.10
2008	42.8%	41.3%	18.73
2009	36.3%	34.7%	28.15
2010	43.1%	41.9%	8.28
2011	40.8%	39.6%	10.10
2012	42.1%	41.0%	5.61
2013	39.0%	37.9%	9.94
2014	39.8%	38.8%	8.38
2015	43.5%	42.6%	18.93
2016	45.6%	44.7%	29.08
2017	48.5%	47.6%	36.64
2018	49.1%	47.9%	34.25
2019	46.4%	45.5%	29.77
2020	48.7%	47.9%	43.61
Mean	42.9%	41.7%	21.08
Median	42.8%	41.3%	18.93

Table 6. Patent Value Benchmark Comparison

This table compares the mean and median performance, calculated across all years, of the full model from Table 5, a gradient boosting regressor trained on the full list of features, and a neural network trained with all features except the embedding of the patent text. I report R-squared score, adjusted R-squared, and mean squared error.

		R^2	Adj. R^2	MSE
Full Model	Mean	42.9%	41.7%	21.08
	Median	42.8%	41.3%	18.93
XGBoost	Mean	28.5%	27.0%	26.87
	Median	28.5%	27.3%	23.95
No Embed	Mean	17.4%	17.4%	31.21
	Median	17.6%	17.6%	27.60

Table 7. Primary Citation Prediction Results

This table reports the year-by-year results of the prediction model for the increase in patent citations from the third year to the tenth year, using machine learning and the full list of features, including the embedding of the patent text. The sample is restricted conditional on the patent receiving at least a single citation within a three-year horizon from publication, which is discussed above. I report R-squared score, adjusted R-squared, and mean squared error, as well as overall mean and median for all statistics.

Year	R^2	Adj. R^2	MSE
2004	29.0%	26.4%	4.65
2005	25.9%	22.6%	4.78
2006	24.9%	22.0%	5.05
2007	23.5%	20.2%	5.14
2008	21.0%	17.6%	5.46
2009	21.9%	18.8%	5.45
2010	22.6%	20.3%	5.69
2011	20.6%	18.1%	6.00
2012	17.4%	15.1%	6.30
Mean	23.0%	20.1%	5.39
Median	22.6%	20.2%	5.45

Table 8. Patent Citation Benchmark Comparison

This table compares the mean and median performance, calculated across all years, of the full model from Table 7, a gradient boosting regressor trained on the full list of features, and a neural network trained with all features except the embedding of the patent text when predicting citation increase from the third year to the tenth year after publication. I report R-squared score, adjusted R-squared, and mean squared error.

		R^2	Adj. R^2	MSE
Full Model	Mean	23.0%	20.1%	5.39
	Median	22.6%	20.2%	5.45
XGBoost	Mean	17.6%	14.6%	5.80
	Median	17.4%	14.5%	5.84
No Embed	Mean	8.0%	7.9%	6.47
	Median	8.2%	8.1%	6.43

Table 9. Application Screening

This table documents the phenomena of "application screening," in which firms adjust and improve lackluster applications in order to secure their ultimate success. Panel A reports the subsample of applications in which a major change in text is recorded between the application and patent grant (defined as a cosine distance of 0.5), and Panel B the applications with observable changes (defined as a cosine distance of 0.2). The predictive model is run on both the application and patent text, and the improvement is reported. The t-statistics for the mean and median improvements are also reported. The t-statistics are reported in parentheses. ***, **, * denote statistical significance at the 0.01, 0.05, and 0.10 levels, respectively.

Panel A: Cosine distance between patent and application at least 0.05.

	Mean	SD	Min.	25 Pct.	Median	75 Pct.	Max.	N
Predicted Success Rate (Application Text)	17.4%	7.7%	0.7%	9.8%	21.4%	23.3%	27.4%	42
Predicted Success Rate (Patent Text)	27.7%	19.3%	3.9%	13.5%	23.4%	35.9%	83.2%	42
Improvement	10.3%***	17.0%	-10.0%	-0.5%	4.1%***	17.3%	61.1%	42
	(3.943)				(3.382)			

Panel B: Cosine distance between patent and application at least 0.02.

	Mean	SD	Min.	25 Pct.	Median	75 Pct.	Max.	N
Predicted Success Rate (Application Text)	18.8%	7.9%	0.7%	13.5%	21.9%	24.1%	38.9%	92
Predicted Success Rate (Patent Text)	25.6%	15.7%	3.9%	13.9%	23.8%	30.0%	83.2%	92
Improvement	6.7%***	13.2%	-10.0%	-0.6%	1.5%***	10.2%	61.1%	92
	(4.920)				(4.381)			

Table 10. ChatGPT Application Revision

This table documents the ability of ChatGPT to "revise" patent applications for greater chance of ultimate success. The AI is given the worst 50 applications from each year, in terms of embedding-only prediction (Initial Text Quality, or *Application Quality* per Section 3.2 and Table 4), and asked to revise for maximum chance of success. Then, the embedding-only model is run on the revised text, producing the measure ChatGPT Revised Quality (i.e., the *Application Quality* of the revised text), and the two are compared. The t-statistics for the mean and median improvements are reported in parentheses. ***, **, * denote statistical significance at the 0.01, 0.05, and 0.10 levels, respectively.

	Mean	SD	Min.	25 Pct.	Median	75 Pct.	Max.	N
Initial Text Quality	8.0%	7.4%	0.1%	3.1%	5.2%	8.3%	25.9%	800
ChatGPT Revised Quality	12.8%	10.1%	0.0%	4.9%	9.8%	20.7%	64.3%	800
Improvement	4.8%*** (18.195)	7.5%	-5.4%	0.4%	2.8%*** (19.808)	6.7%	58.2%	800

Table 11. Comparison of AI-based Value Measure with KPSS

This table reports the deviation of the AI-assisted measure of patent value from the KPSS value. I define Adjusted KPSS as the KPSS measure scaled assuming a blanket factor of $p_1 = 72.4\%$ acceptance rate, rather than $p_0 = 55\%$ acceptance. The new, AI-revised measure is defined as $KPSS \cdot 1/(1-\hat{p})$, where $\hat{p}$ is the predicted chance of success from the application model. The first two rows show the differences in scaling factor, i.e., $\frac{1/(1-\hat{p})}{1/(1-p_0)}$, etc. while the remaining rows show the summary stats of and comparison between the estimated value metrics in millions of dollars, adjusted for inflation.

	Mean	SD	10 Pct.	25 Pct.	Median	75 Pct.	90 Pct.	N
Scaling Comparison								
AI/KPSS	3.65	3.78	0.58	1.45	2.46	4.31	24.56	757,560
AI/Adj. KPSS	2.24	2.32	0.36	0.89	1.51	2.65	15.07	757,560
Value Comparison ($M)								
AI	30.30	70.89	0.27	1.59	8.10	25.87	72.25	481,152
KPSS	8.75	17.49	0.09	0.56	2.88	8.69	22.11	481,152
Adj. KPSS	13.58	27.13	0.15	0.86	4.46	13.48	34.31	481,152
AI - KPSS	21.55	59.35	0.03	0.43	3.79	15.81	49.87	481,152
AI - Adj. KPSS	16.72	54.35	-1.44	0.04	1.77	11.18	39.78	481,152

Table 12. AI-adjusted Patent Values and Forward Citations

This table reports the results of the comparison between the AI-based value metrics, KPSS value, and forward citations. Forward citations is defined as the logarithm of 1 plus all citations over a three-year horizon from the grant date. Three AI-based value metrics are defined as follows. Predicted success rate, or $\hat{p}$, is the predicted chance of success for a certain application of the primary application machine learning model. The AI scaling factor is defined as $1/(1-\hat{p})$. The AI-adjusted value is the product of KPSS value and the AI scaling factor (the KPSS value is first descaled from the original constant scaling factor. This is irrelevant in the context of linear tests). Panel A shows the results of the regression of value metrics on forward citations, as per Table 2 of KPSS. Panel B reports the correlation matrix for the four metrics. Panel C reports multilinear regressions of citations on, respectively, AI scaling factor with KPSS value and predicted success rate with KPSS value. T-statistics are reported in parentheses, with standard error clustered at the firm level. ***, **, * denote statistical significance at the 0.01, 0.05, and 0.10 levels, respectively.

Panel A: Regression of Value Metrics on Citations.

	(1)	(2)	(3)	(4)
Dependent Variable	KPSS Value	AI-adjusted value	AI Scaling Factor	Predicted Success Rate
log(1+$C_{j,t}$)	0.128***	1.556***	0.490***	0.005***
	(2.71)	(3.16)	(4.78)	(5.84)
Constant	-33.583**	-165.881**	2.939	0.680***
	(-2.55)	(-2.23)	(0.23)	(13.10)
Patent Controls	Y	Y	Y	Y
Firm Size Control	Y	Y	Y	Y
Year Fixed Effects	Y	Y	Y	Y
Firm Fixed Effects	Y	Y	Y	Y
N	451,131	451,131	451,131	451,131
Adj. R^2	0.611	0.179	0.092	0.272

Panel B: Correlation Matrix of Value Metrics.

	AI-adjusted Value	KPSS Value	AI Scaling Factor
KPSS Value	0.503		
	(0.00)		
AI Scaling Factor	0.245	-0.036	
	(0.00)	(0.00)	
Predicted Success Rate	0.121	-0.092	0.374
	(0.00)	(0.00)	(0.00)

Panel C: Regression of Citations on Value Metrics.

	$\log(1 + C_{j,t})$	
	(1)	(2)
AI Scaling Factor	0.001***	
	(3.79)	
Predicted Success Rate		0.231***
		(5.20)
KPSS Value	0.001***	0.001***
	(2.81)	(2.88)
Constant	0.106	-0.049
	(0.92)	(-0.38)
Patent Controls	Y	Y
Firm Size Control	Y	Y
Year Fixed Effects	Y	Y
Firm Fixed Effects	Y	Y
N	451,131	451,131
Adj. R^2	0.106	0.107

Table 13. Application Strength and Portfolio Returns

This table reports the monthly performance of the long-short portfolio. The firm-level measure, *application strength*, is defined as the mean predicted chance of acceptance for all applications of a firm published in a given month, multiplied by the square root of the number of applications. The firms are sorted into two groups, above and below median, based on *application strength*, every month. A long-short portfolio is then constructed for the two groups with a one-month horizon. Value-weighted abnormal monthly returns are calculated based on the Fama-French three-, four-, and five-factor models. Significance is determined based on the one-tailed t-statistic test, and the t-statistics are reported in parentheses. ***, **, * denote statistical significance at the 0.01, 0.05, and 0.10 levels, respectively.

	FF3-adjusted Return	FF4-adjusted Return	FF5-adjusted Return
Low Application Strength	-0.144%*	-0.182%*	-0.151%*
	(-1.30)	(-1.58)	(-1.36)
High Application Strength	0.081%*	0.094%*	0.086%*
	(1.31)	(1.50)	(1.31)
Difference (High - Low)	0.235%**	0.276%**	0.237%**
	(1.78)	(2.10)	(1.84)

Internet Appendix for
"Predictive Patentomics: Forecasting Innovation Success and Valuation with ChatGPT"

This Internet Appendix provides additional figures and tables supporting the text.

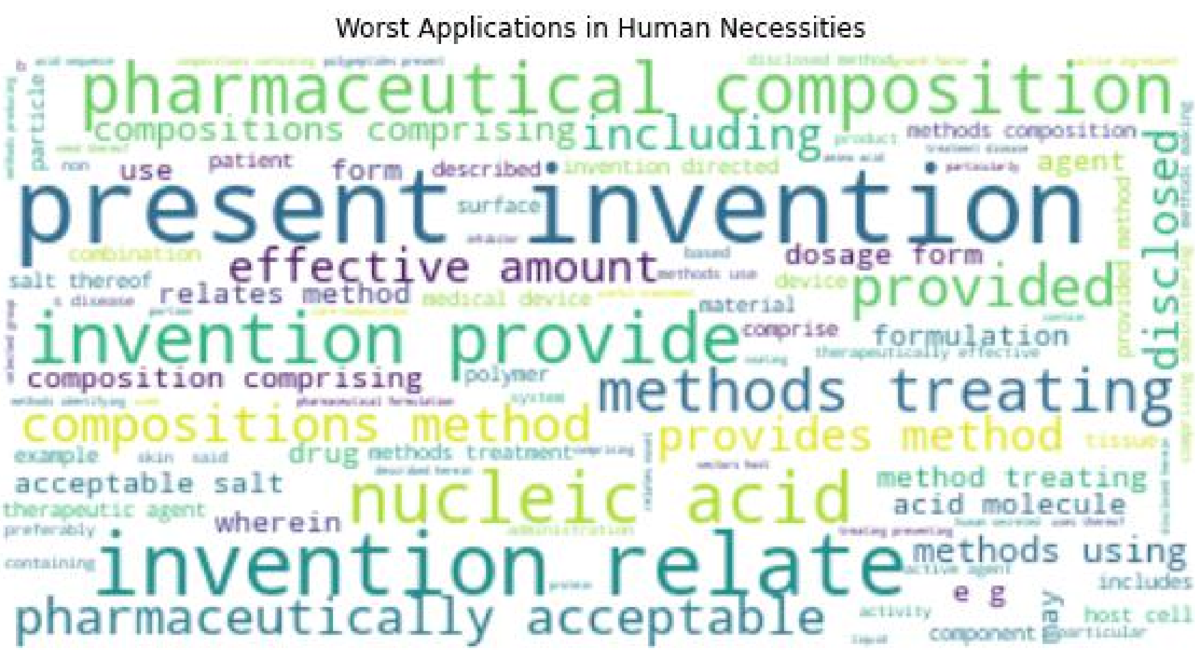
- Figure IA.1 provides additional world clouds for the best and worst applications, supplementing Figure 3.
- Table IA.1 extends the tests in Table 6 to 2022, for out-of-sample (OOS) validation of performance and to address potential look-ahead bias.

Figure IA.1. Additional Word Clouds of Best and Worst Applications

This figure documents the “word clouds” for the title and abstracts of the 30,000 worst and best applications (defined as having, respectively, the lowest and highest predicted chance of success by the full model) for major patent categories in addition to those shown in Figure 3.

Panel 1A: Best in Human Necessities.

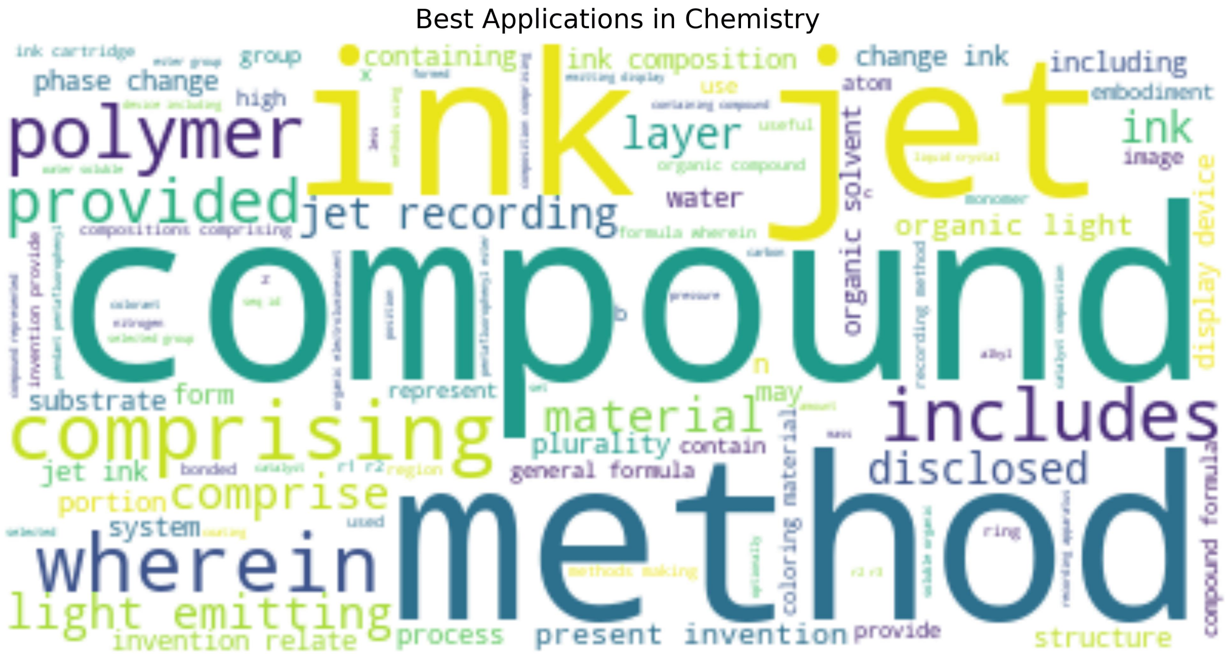
Panel 1B: Worst in Human Necessities.



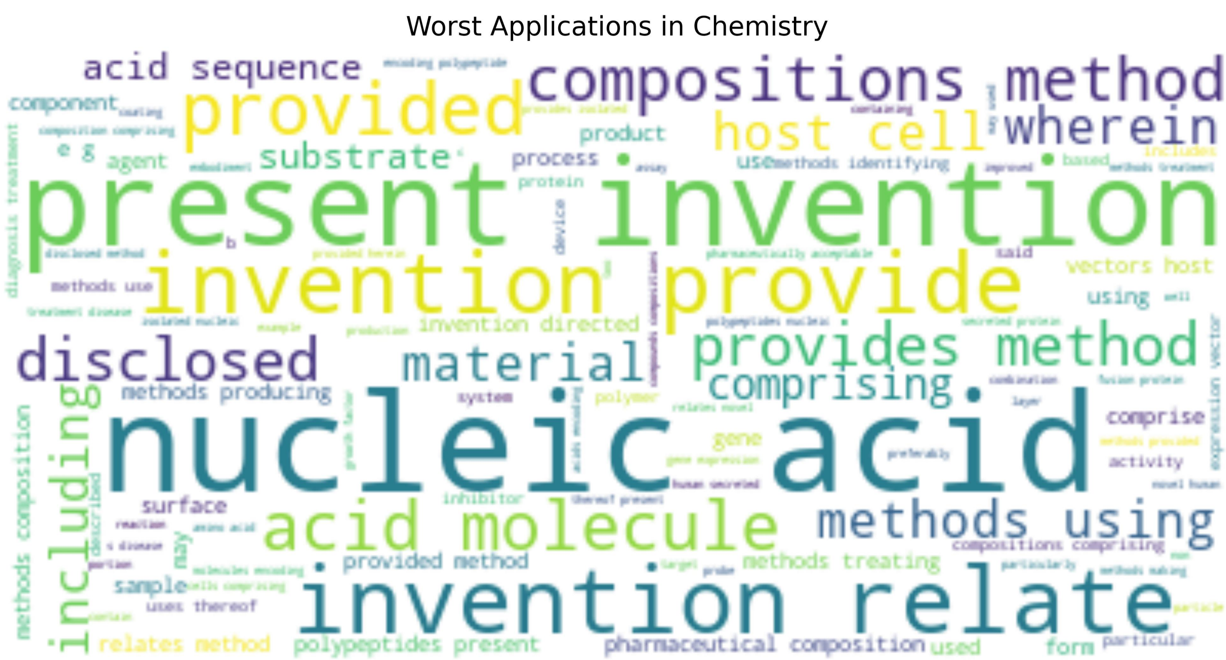
Panel 2A: Best in Operations.

Panel 2B: Worst in Operations.

Panel 3A: Best in Chemistry.



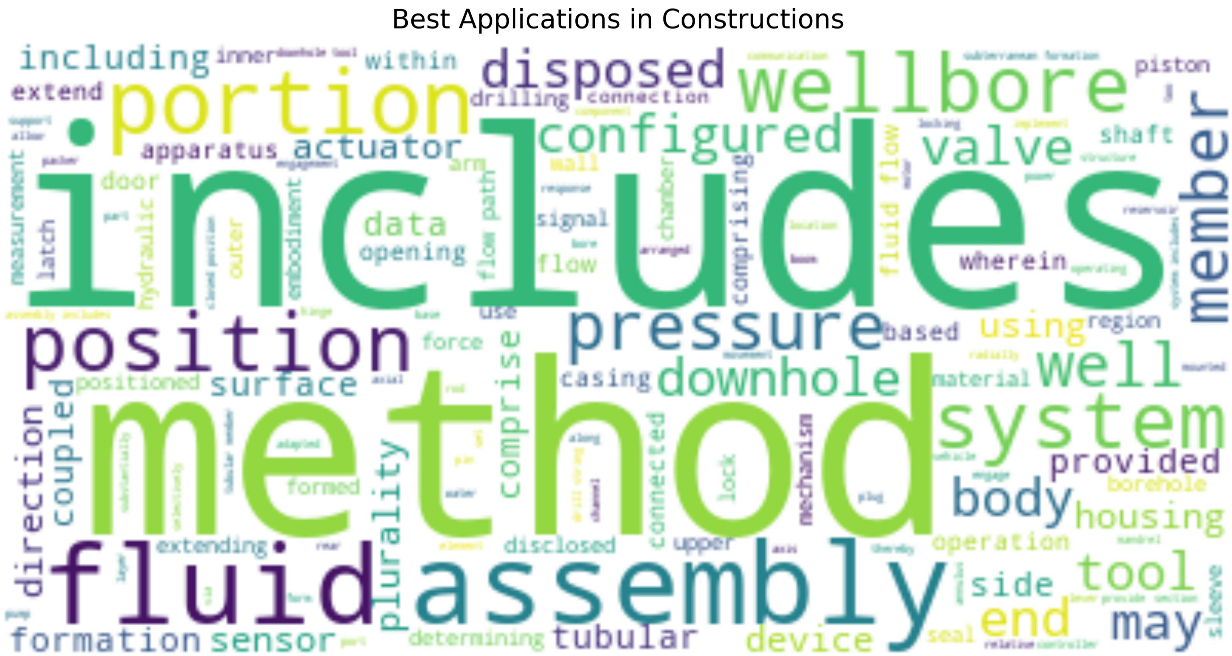
Panel 3B: Worst in Chemistry.



Panel 4A: Best in Textiles.

Panel 4B: Worst in Textiles.

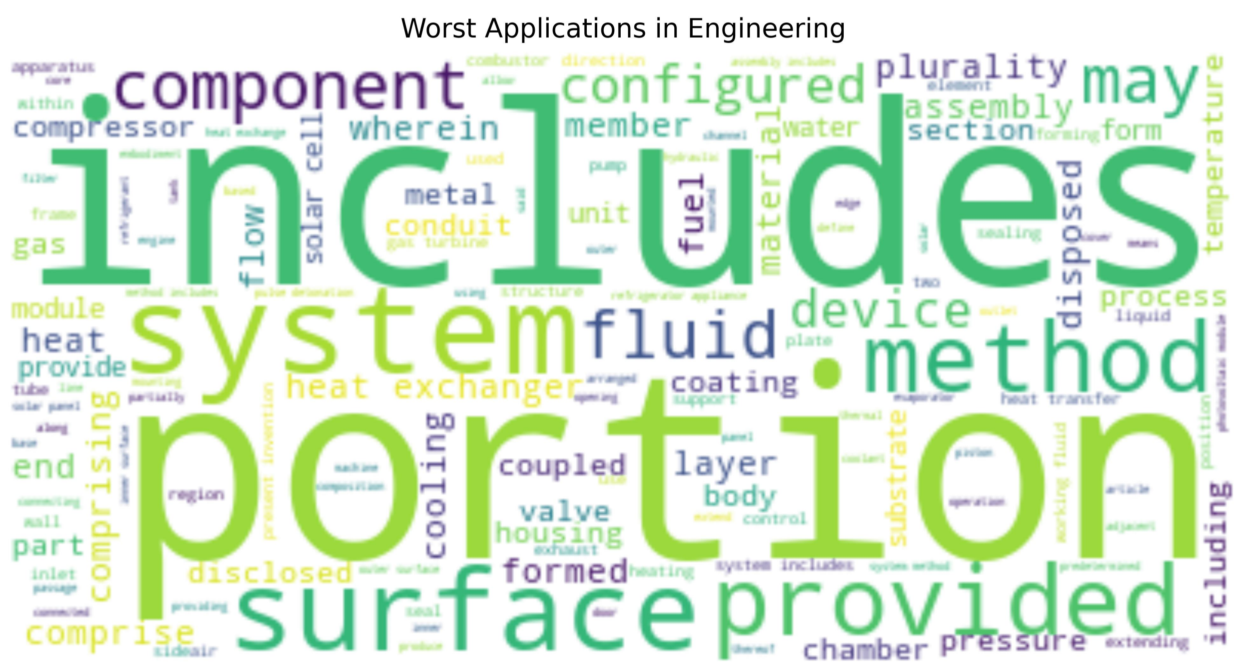
Panel 5A: Best in Constructions.



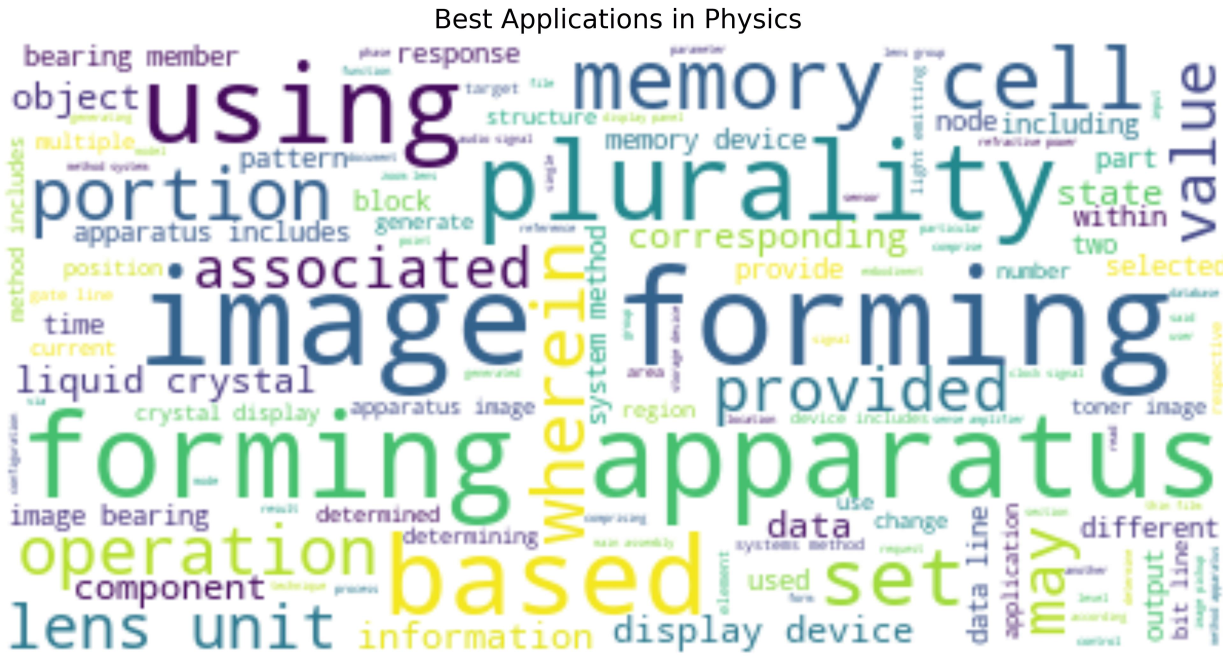
Panel 5B: Worst in Constructions.

Panel 6A: Best in Engineering.

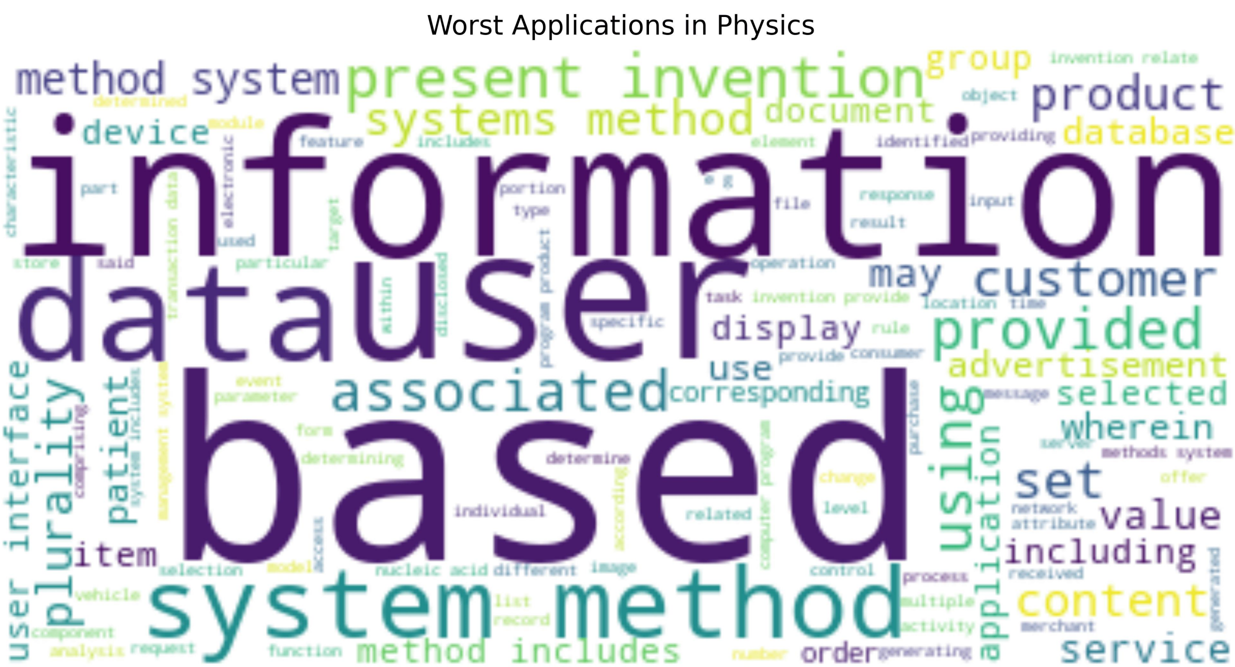
Panel 6B: Worst in Engineering.



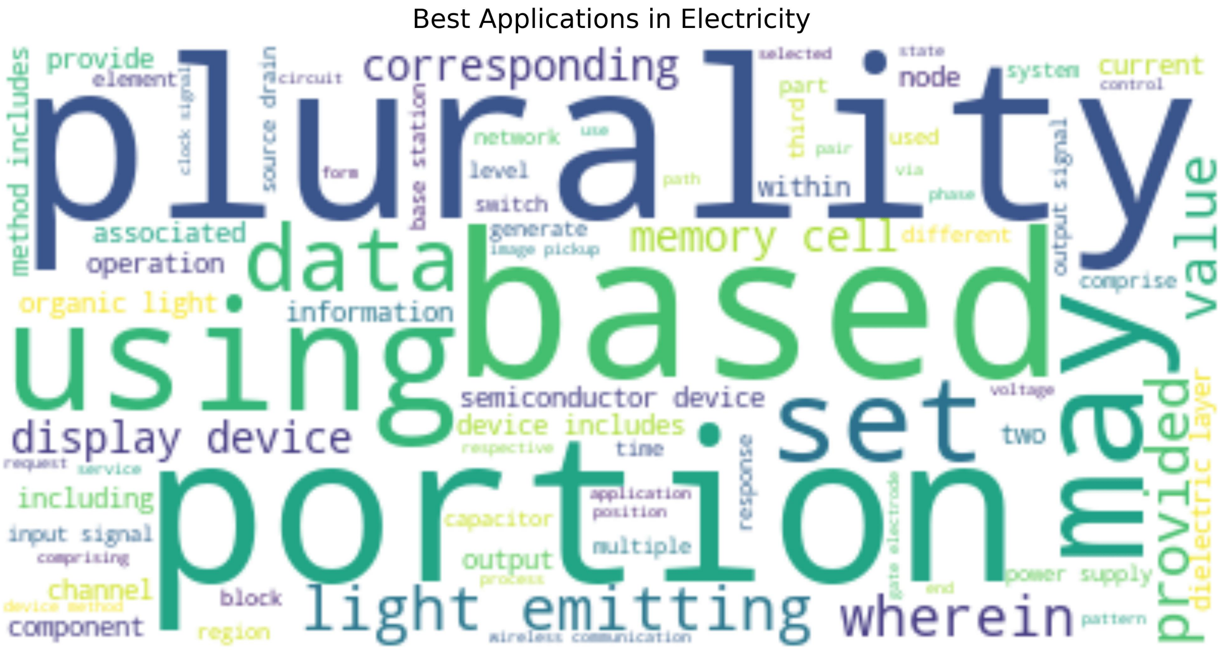
Panel 7A: Best in Physics.



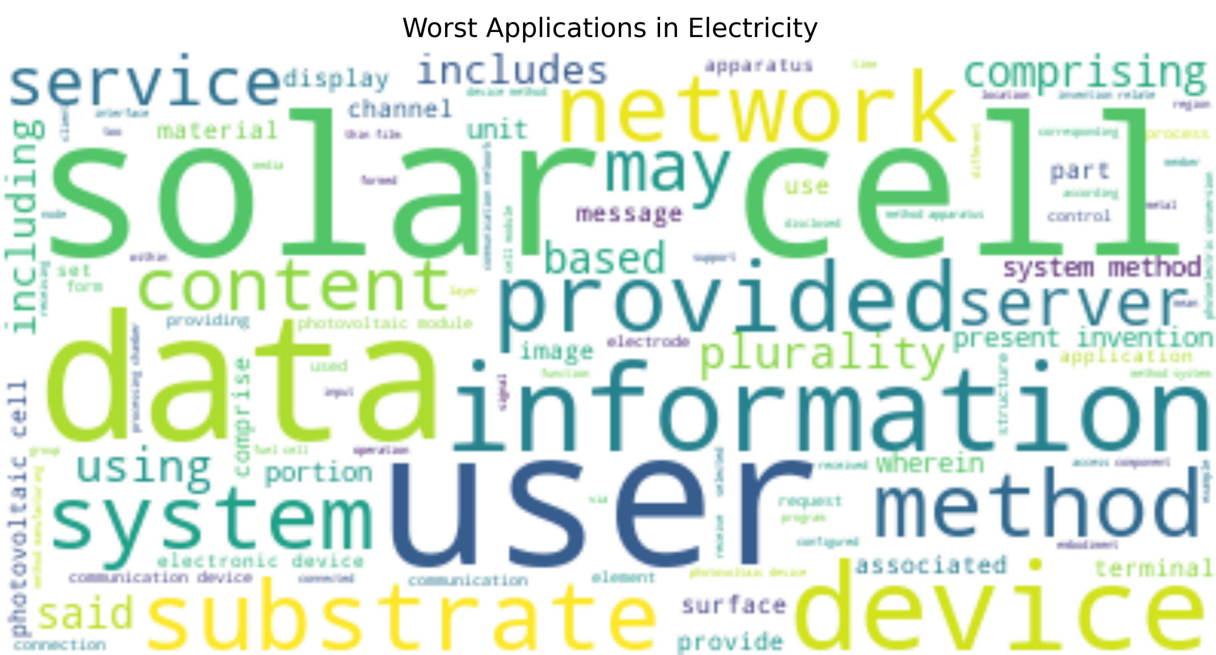
Panel 7B: Worst in Physics.



Panel 8A: Best in Electricity.



Panel 8B: Worst in Electricity.



Panel 9A: Best in High Tech.

Panel 9B: Worst in High Tech.

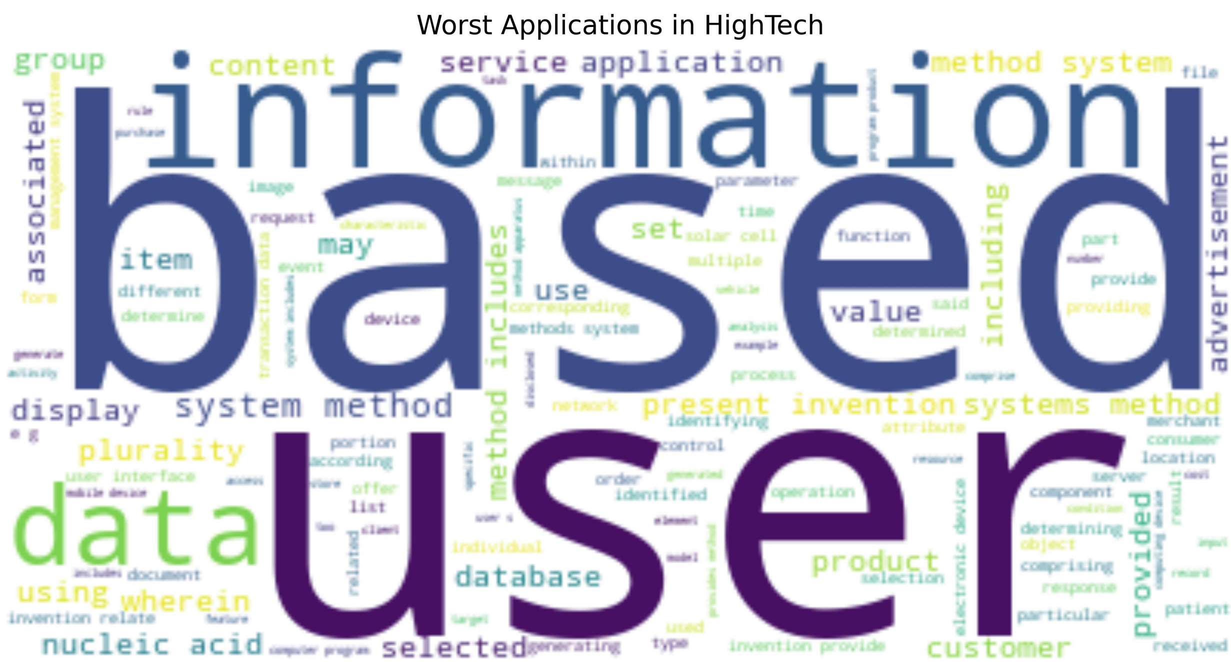


Table IA.1. Patent Value Predictions: Out-of-sample Tests in 2022

This table compares the performances of the full deep learning model for patent value prediction, a gradient boosting regressor trained on the full list of features, and a neural network trained with all features except the embedding of the patent text, for out-of-sample tests in 2022. I report R-squared score, adjusted R-squared, and mean squared error.

	R^2	Adj. R^2	MSE
Full Model	43.1%	42.4%	10.16
XGBoost	35.1%	34.3%	11.60
No Embed	24.2%	24.2%	13.54