AI and Bond Values: How Large Language Models Predict Default Signals

Moazzam Khoja

Central Washington University

Version Date: September 26, 2024

Abstract

This paper investigates the potential of Large Language Models (LLMs) to interpret textual data from company earnings calls and assess default likelihood. Using ChatGPT to analyze earnings call transcripts, I find that LLM-derived default likelihoods enhance the predictability of corporate bond credit spreads, independent of stock prices and liquidity measures. By comparing actual credit spreads with counterfactual spreads predicted using LLM-based default likelihoods, I show that LLMs reduce serial correlation in credit spreads, indicating less under-reaction to information. This research highlights LLMs' role in improving market efficiency by providing a consistent method for processing textual data.

1 Introduction

Financial information comes in two forms: structured data, like analysts' median forecasts or actual quarterly earnings, and textual data, such as management's business descriptions or transcripts of discussions with analysts. Agents use this information to form rational expectations about a company's health. However, textual information is more prone to noise and interpretational dispersion than structured data. On an individual level, noise arises from biases due to priors, selective attention, and imperfect comprehension [Coibion and Gorodnichenko (2012), Angeletos et al. (2021)], while interpretational dispersion occurs due to the varying perspectives of different agents.

Large Language Models (LLMs) offer a promising technology that applies consistent contextualization to text. As algorithms, they can mitigate prior beliefs and reduce selective attention. Their uniform application and consistency help decrease interpretational dispersion. While LLMs may not entirely eliminate human biases—since they are trained on human-generated text—they can still reduce some biases [Chen et al. (2023)]. This paper uses LLMs to interpret company earnings calls, formulating rational beliefs about a company's default likelihood, and tests these beliefs against corporate bond credit spreads.

The paper's key insights reveal that LLM-formulated default likelihoods offer an additional predictive signal for credit spreads. Stock prices remain unaffected by default likelihoods, and there are no spillovers from stocks to bonds. In the second part of the paper, I use a machine learning algorithm to predict out-of-sample credit spreads based on LLM-derived default likelihoods. These predicted spreads represent a counterfactual scenario—a hypothetical world where agents use LLM-based beliefs to set prices. The paper demonstrates that the serial correlation in the counterfactual credit spreads is eliminated while it exists in the actual data. From this, I conclude that if LLMs had existed during the sample

period, they would have reduced the under-reaction to information seen in the actual data, thus providing a tool to make markets more efficient.

LLMs are trained on large corpora of text to model the statistical properties of language, with a particular focus on understanding the context in which each word appears. In these models, each word is associated with three key vectors: the query vector, which seeks contextual information from surrounding words; the key vector, which represents the relevance of other words in the sentence; and the value vector, which encapsulates the actual content or meaning of the word. These vectors work together to refine the contextual representation of each word within the sentence. The purpose of this architecture is to enhance the model's ability to capture the nuanced relationships between words, thereby improving its overall comprehension. With billions of parameters fine-tuned through extensive training, LLMs can generate sophisticated contextual representations, offering a significant advancement over earlier models that relied primarily on simpler techniques like sentiment analysis or word frequency counts [Brants et al. (2007)]. This advanced capability allows LLMs to uncover subtle patterns in language, making them particularly adept at interpreting complex interactions, such as those found in management-analyst Q&A sessions during earnings calls.

ChatGPT (an LLM) interprets rational beliefs that closely mimic human beliefs, as demonstrated by Bybee (2023). He extracted ChatGPT-formulated beliefs from macroeconomic data and validated them against actual surveys. Statistically, LLM predictions lower the mean squared error in estimating the truth by reducing variance. The benefits of this error reduction have been empirically verified; for instance, Kim et al. (2024) shows that ChatGPT outperforms human analysts in predicting earnings, even when provided with only tabular data. This paper builds on this research by extracting insights from textual conversations on more abstract concepts, such as default likelihood.

This paper uses summarized historical earnings call transcripts as input to ChatGPT 3.5, prompting it to score the likelihood of a company's default on a scale of 1-10 (*default likelihood*) based on the conversation in the transcript. It then examines the impact of this default likelihood on the company's credit spreads because credit spreads contains market implied default probabilities [Chan-Lau (2006)]. The findings reveal that ChatGPT's interpreted default likelihood has significant explanatory power in predicting credit spread movements. This relationship holds even after controlling for factors such as liquidity measures, ratings fixed effects, and firm fixed effects. Additionally, the analysis shows that the full model incorporating ChatGPT's default likelihood is significantly different from a restricted model without it, demonstrating that this approach adds explanatory power beyond what non-textual information provides.

Several studies have used artificial intelligence to predict defaults [Donovan et al. (2021), Siddiqui et al. (2023), Wu et al. (2023), Kriebel and Stitz (2022)]. However, this paper takes a different approach: rather than using LLMs for outcome prediction, it employs them to assess rational beliefs and then evaluates how these assessments contribute to explanatory power. Importantly, the assessment process is separate from the determination of explanatory power. The goal is not to build a predictive model, at least in the first test, but to investigate whether these signals influence asset prices. This paper explores how LLMs contribute to improving market efficiency.

The impact of information on bond prices is not thoroughly explored in the literature. One reason is that, unlike earnings announcements that directly affect stock prices, there is no specific event that provides a clear release of information about a company's default likelihood. This paper offers a unique opportunity to examine the effect of LLM-interpreted default likelihood and the persistence of market reactions over time. I fit a vector autoregression (VAR) model on the residuals of credit spreads—after accounting for ratings, firm

fixed effects, and liquidity measures—alongside default likelihood and stock prices. This VAR tests whether default likelihood influences stock prices and credit spreads, and it also examines spillover effects and persistence over time. The results show no spillover effects or impact of default likelihood on stock prices. However, credit spreads do show a short drift of one meeting despit no lagged effect from prior default likelihood. The result indicates an under-reaction to information. These findings are consistent with Gebhardt et al. (2005), who demonstrated a reversal in bond prices, implying positive serial correlation in credit spreads.

Another issue discussed in the literature is the cyclicality of credit spreads. Greenwood and Hanson (2013) examine how economic boom and bust cycles affect credit spreads, finding that spreads narrow during boom periods. They attribute this narrowing to behavioral factors, suggesting that credit markets become "overheated" during booms, leading to lower credit quality. They tested this hypothesis by identifying an increased proportion of low-quality credit during boom years (2004-2007). In Greenwood et al. (2019), a theoretical model is proposed to explain this phenomenon. However, a limitation of empirical tests based on behavioral theories is controlling for issuer-specific information. It is possible that default likelihood improves during boom times, potentially justifying the overheating effect.

A key advantage of using LLM-based default likelihood is the ability to test whether credit spreads narrow despite high default likelihood during boom years. My findings are inconclusive: during boom years, default likelihood has a negative coefficient with credit spreads. Despite a high likelihood of default based on rational assessments of a company's health, credit spreads remain narrow during booms. However, the effect is not statistically signifiant. We cannot rule out whether narrowing of credit spreads ignores rational expectation based on company's health during that time.

The literature on Post Earnings Announcement Drift (PEAD) is extensive. Bernard and Thomas (1989) offers a thorough empirical analysis, suggesting that the most likely cause is the time required to process and diffuse information, which leads to an under-reaction in stock prices to the news. Barberis et al. (1998) and Daniel et al. (1998) provide behavioral explanations for PEAD, while Hong and Stein (1999) attribute it to segmented agents. Most studies reject changes in risk expectations as a cause [Jegadeesh and Titman (1995), Jegadeesh and Titman (1993)].

Although this paper focuses on corporate bonds, similar characteristics are observed in credit spreads, such as serial correlation and lagged effects of default likelihood. This suggests that under-reaction to information might also be prevalent in credit spreads. If we assume that, similar to PEAD, the market's inability to fully absorb and diffuse information is causing these patterns, a way to test this hypothesis is by creating counterfactual data on credit spreads using machine learning algorithms. A similar approach is used by Van Binsbergen et al. (2023), who forecast earnings with machine learning algorithms and compare them to analyst forecasts to examine biases and their implications for stock returns and managerial decisions. In this paper, Random Forest regression is employed to forecast credit spreads.

Random Forest regression is a supervised machine learning algorithm that builds multiple decision trees using bootstrapped samples (sampling with replacement). Each tree in the forest is trained on a different subset of the data and features. The final prediction is based on the average predictions from all the trees in the forest.

During training, Random Forest adjusts the features and their interactions to minimize mean squared error. One of its strengths is handling complex interactions and high-dimensional feature sets effectively[Van Binsbergen et al. (2023)]. It achieves this by considering only a random subset of features for each split, which promotes diversity among the

trees and helps prevent overfitting. After training, the model is validated with an out-of-sample validation set.

I use several regressors—including credit ratings, company fixed effects, liquidity measures, and ChatGPT scores—to predict credit spreads. The hypothesis is that if under-reaction results from the processing and diffusion of information, then incorporating LLM-based information processing should reduce under-reaction. A simple test would be to compare the serial correlation of predicted credit spreads with that of actual spreads.

The results show that actual credit spreads exhibit high autoregressive coefficients, while the coefficients for predicted credit spreads have no auto-regressive coefficient. This suggests that if technology to assess textual information had been available, it would have reduced the effects of under-reaction to information. Since autoregression in actual spreads does not incorporate textual information, it contrasts with the predicted spreads, which do. This indicates that LLMs reduce frictions in information processing and diffusion, thereby mitigating under-reaction to information. The result also show that even when default likelihood is a sole predictor in OLS, the auto-regressive coefficients of predicted spreads reduce considerably compared to autoregression of observed spreads, providing evidence that LLMs contribute to reducing PEAD.

This paper contributes to several areas of research. First, it builds on an emerging body of literature exploring the use of LLMs to understand stock prices, corporate policies, and other aspects [Kim et al. (2023b), Bernard et al. (2023), Chen et al. (2022), Eisfeldt et al. (2023), Kim et al. (2023a), Jha et al. (2024)]. These studies demonstrate the value of LLMs in improving predictions and understanding outcomes and corporate decisions. This paper extends this work by applying LLMs to assess default likelihoods.

Second, to the best of my knowledge, the impact of corporate news on corporate bond values has not been previously studied. This paper provides new empirical insights into the impact of news on bond values. It further explores cyclical nature of bond values during boom and bust periods, however it cannot rule out rational explanation of market exuberance during boom years.

Lastly, this paper advances the literature on the causes of under-reaction to news. It uses machine learning tools to show that these tools can provide better predictive power to reduce under-reaction to the news.

The rest of the paper is organized as follows: Section 2 provides details on LLMs and Random Forest, Section 3 outlines the data sources and extraction process, Section 4 presents the results, and Section 5 concludes the paper.

2 Technical Details

2.1 Large Language Models

The use of textual analysis to predict effect on asset pricing has been studied [Baker and Wurgler (2007),Bybee et al. (2023),Manela and Moreira (2017)], However LLMs ability to think differently from focus on words was a major breakthrough in understanding meaning in sentences.

LLMs use a model architecture called "Transformer," which was introduced by researchers at Google [Vaswani et al. (2017)]. This architecture was revolutionary due to its use of attention mechanisms to process sequences of data. The process begins by representing each word as a vector of values. Initially, these vectors, known as embeddings, do not have context

within a sentence. For example, the vectors representing the words "red" and "blue" might have a high degree of similarity in isolation. This similarity is often measured using the dot product of these vectors, cosine similarity of these vectors.

However, the transformer architecture applies contextualization to understand the meaning of words within sentences. For instance, while "red" and "blue" might be similar in isolation, in the sentence "A genre of music, blues, is about sorrows," the word "blue" refers to a musical genre rather than the color. In this context, the vector for "blue" will differ significantly from its vector in other contexts. The transformer modifies the embeddings based on the context provided by the surrounding words.

This contextualization is achieved using three types of vectors: query, key, and value.

- Query Vector: Represents the word's need for information from other words in the sentence. It helps determine how the word should interact with other words to understand its meaning. For the word "blues," the query vector seeks information about how "blues" relates to other words in the sentence. It is used to find out what other words can provide useful context for "blues."

- Key Vector: Represents the information or content of other words in the sentence. It helps evaluate how relevant these words are to the query vector. For example, the key vector for "sorrows" helps assess how relevant it is to understanding the meaning of "blues" in this context.

- Value Vector: Represents the meaning of the word based on its context in the sentence. It reflects the final, contextually informed representation of the word. For example, The value vector for "blues" will capture its meaning based on the context provided by the sentence "A genre of music, blues, is about sorrows." This helps in forming the final contextual representation of "blues," indicating that in this sentence,"blues"

refers to a musical genre associated with emotions like sorrow, rather than the color blue.

The transformer model computes an attention score by taking the dot product of the query vector and the key vector for each word. This score is then normalized using a softmax function to produce an attention weight, a scalar that indicates the relevance of each word's value vector to the current word's query vector. The weighted sum of the value vectors of all words in the sequence, based on the attention weights, produces the final contextual representation for each word.

The learning process involves updating the embeddings through training on a large corpus of text. This allows the transformer model to refine its attention weights and improve its ability to understand and generate text based on context.

The transformer architecture equips LLMs with the ability to understand the context of words within a sentence. Consequently, when asked to interpret a transcript between management and analysts, LLMs can discern the context of each word. This contextual understanding enables them to extract meaningful insights from the text. For example, by analyzing the context in which words appear, an LLM might assess the likelihood of default based on the content of the transcript.

The likelihood assessed by the LLM will not be influenced by the company's past records or other external information, as the model is only prompted to analyze a single transcript at a time. The LLM uses a consistent set of underlying weightings to generate scores across all transcripts, meaning it applies the same interpretation method uniformly and does not deviate or get distracted by information outside the transcript. This ensures a consistent application across all companies; for example, the attention given to a technology company will

be consistent with that given to an energy company. Additionally, since a single algorithm evaluates the information, it avoids interpretational dispersion within the same text.

2.2 Random Forest Regression

Random Forest Regression is a type of supervised learning algorithm used to predict an outcome variable, such as "credit spreads," based on a set of input features (regressors). The method is an extension of decision tree regression, where the data is repeatedly split into smaller and smaller subsets based on different features.

2.2.1 Basic Concept

Imagine you create a decision tree that first splits the data based on a feature like credit ratings. The tree divides the data into two groups: one with ratings above investment grade and one with ratings below investment grade. Within each group, the tree might further split the data based on another feature, such as the specific credit rating (e.g., splitting between BBB- and BBB+). For each subset of data, the model then uses a regression to predict credit spreads. The final prediction from this single decision tree is derived from these splits and the regression outcomes.

2.2.2 Random Forest

Random Forest takes the idea of decision trees to a more powerful level. Instead of building a single decision tree, Random Forest constructs a large number of trees, each trained on a random subset of the data and using a random subset of the features. For example, one tree might focus on below-investment-grade ratings and further split data based on the average dollar amount traded (e.g., between \$100M and \$200M, or below \$5M). Each tree in the forest makes its own prediction, and the final prediction from the Random Forest model is the average (or mean) of all the individual tree predictions.

The benefit of Random Forest regression lies in its ability to enhance diversity among features by randomly selecting subsets of them. This diversity reduces the risk of overfitting by ensuring that the model does not overly rely on any single feature. Instead, the errors from individual trees, which might overfit, tend to cancel each other out. As a result, the model generalizes better and performs well on out-of-sample data.

3 Data

The data generation process starts with obtaining all companies for which bond data exists in compustat. I remove all bonds that are issued by subsidiaries or special purpose entities by eliminating all tickers that contain an extension after "." in the symbol. The final list contains 945 ticker symbols for which bonds exists in the compustat database.

From this list, I find monthly bond prices, outstanding bond notional, and other information from "Bond Returns with WRDS" database from January 2000 to December 2023. The bond return data contains data for the month end. The bond price is based on last traded bond value within the month, therefore it contains valuation based on actual trades rather than models. In addition, it contains other information like dollar value of trades for each bond and bid and offer spreads on trades which is a valuable indicator of liquidity.

Credit spreads represent the percent spread above Treasury yield of similar maturity for each bond. To calculate credit spreads, I obtain constant maturity Treasury yield from FRED (Federal Reserve Bank of St. Louis). I use the constant maturity in days closest to the maturity of each bond to obtain Treasury yield. Credit spreads is calculated as a difference between yield on corporate bond and Treasury yield for the closest maturity. Bad prints like bonds with negative yields or negative credit spreads are removed from the data.

I take the average of credit spreads, bid-to-offer spreads, and other measures of all the bonds of the same company within the month. Therefore I get around 23 years of monthly average credit spreads across 945 companies.

I collect transcripts from S&P Capital IQ. These transcripts are obtained from the list of companies for which I have bond data. I obtained these transcripts from January 2000 to December 2023. These transcripts come in several phases. Preliminary transcripts and then finalized version. However, for this analysis, I use the first available transcripts that hit the market. Each transcript has information about the type of call, for example earnings calls or merger and acquisition. There were 80,296 transcripts in all categories. From these transcripts, I filter transcripts that pertain to earnings calls. These calls are management's discissions with analyst community including question answer sessions that contains the most valuable textual information about the company's prospects and earnings. I do not evaluate transcripts that are related to merger and acquisitions or sales and trading calls amongst other to avoid "promotional" calls. This leads to 49,856 transcripts for 933 companies. From these transcripts I could match transcripts of 899 companies with bond data which gives 36,393 transcripts. Of this transcripts, I obtain default likelihood score from chatGPT on 28,533 transcripts. For the difference, chatGPT could not assess default likelihood from the conversation (see below) representing 899 companies.

I obtain matching ticker's daily stock price data from CRSP returns data. CRSP data is used to calculate the impact of transcript on stock prices which are then added as control in all regressions. I take average stock price from the day after transcript date to the day before the next transcript date for each company.

3.1 Redrafting Transcript

ChatGPT 3.5 has a limitation to process a maximum token size of 6000. This translates on average into 4,600 words. While a typical transcript is of around 9000 words. Therefore, the first step is to break those transcripts that exceed this threshold into smaller "chunks" of no more than 6000 tokens. Next step is to redraft the text removing unimportant information. This process is also done through chatGPTby using the prompt "Redraft the following text to remove unimportant details and focus on key points. Ensure it is concise." Finally, the redrafted texts of transcripts that have more than 6000 token size is assembled again to form a redrafted transcript. Eventually this transcript is fed to a separate prompt: "On a scale of 1-10, how likely is it that the company will default based on the following transcript? If there is insufficient information to determine, respond with 99. Only provide the number as the response and nothing else." Example of a full transcript and its redrafted version is provided in the internet appendix.

4 Results

4.1 Summary Statistics

The data consists of credit spreads from 899 companies. Around 23,000 observations with bond information and transcripts are obtained. However, 18,790 observations have a valid default liklihood meaning that chatGPT was unable to assess default likelihood of about 7,000 transcripts. The average market capitalization of the sample is $38 billion, though this figure is skewed by very large companies. The median company's market capitalization is around $10 billion. The bonds have an average maturity of approximately 8 years. The average market value of outstanding bonds for companies in the sample is $54 million which is closer to median value about $45 million. Approximately 60% of the observations are for investment-grade bonds, while 40% are high-yield bonds. Credit spreads range from 0.003%

to 99.82% with a median credit spread of1.62%. This is highly skewed with the mean credit spread of 2.61%. A few very large credit spreads are included because they may represent bonds near default with very low recovery potential.

ChatGPT assessed default likelihood on a scale from 1 to 10, with an average likelihood of 4 and a median value of 3, indicating that most transcripts do not predict an imminent default. However, the standard deviation of 2 shows that default likelihood varies significantly depending on the conditions discussed in the transcripts.

4.2 Default Likelihood Effect on Credit Spreads

The credit spread prediction is shown in Table 2. The first column presents results for the full sample. Credit spreads are calculated from the last trade of each month. I use stock returns from the day after the meeting until the last day before the final bond valuation to control for information spillover from stocks that might influence credit spreads. Additionally, I include the bid and offer spread of bonds traded to account for liquidity and its potential impact on credit spreads. I also control for maturity to capture duration-related effects. Beyond these controls, I include ratings fixed effects to account for variations in creditworthiness, and firm fixed effects to account for firm-specific factors that may affect credit spreads during the observation period.

The results show a significant effect on the direction of credit spreads. For each 1-unit increase in the default likelihood, credit spreads are associated with a 0.034% increase. This demonstrates a clear direct relationship between default likelihood and credit spreads

The effect is evident in both investment-grade and high-yield bonds, as shown in columns 2 and 3 of the table. High-yield credit spreads are less sensitive to ChatGPT-assessed default likelihood than investment-grade spreads. For each 1-point increase in default likelihood,

credit spreads move by 0.047% for investment-grade bonds, but by 0.02% for high-yield bonds. This could be because high yield bonds may be trading more like a small cap stocks which may be less sensitive to default likelihood.

The last column of Table 2 presents the test of whether default likelihoods are less effective during boom years. The boom years are defined as the period from 2004 to 2007, consistent with Greenwood and Hanson (2013). The results show an interaction term of -0.087% during boom years, indicating that default likelihood has a negative relationship with credit spreads in that period. Despite ChatGPT-assessed low default likelihood in boom time and narrrowing ofcredit spreads during the boom years, it cannot be conclusively determined that this effect is purely behavioral because the interaction term is not statistically significant. The statistical insignificance mean that it could not be conclusively confirmed that the "overheating" is the only cause. If the effect were statistically significant, it could be concluded that despite high default likelihood, spreads narrowed in boom years.

Table 3 presents the results of an ANOVA and Wald test on two models. The first model is the full model, which includes default likelihood along with the controls and fixed effects mentioned earlier. The second model is a restricted model that excludes default likelihood. The purpose of the test is to determine whether the full model is significantly different from the restricted model—in other words, whether default likelihood significantly improves the predictability of credit spreads. The results clearly indicate that default likelihood makes a significant contribution to predicting credit spreads.

4.3 Time Series Analysis

Table 4 presents the results of a vector autoregression (VAR) analysis, where the variables are the residuals of credit spreads, default likelihood, and stock returns. These residuals are obtained from ordinary least squares (OLS) regressions of the respective variables on ratings fixed effects, firm fixed effects, maturity, and the dollar value of traded bonds. This approach isolates the residual effects on these variables once firm-specific, bond-specific, and credit-specific factors are accounted for. The goal of this regression is to examine whether these residuals exhibit persistence or if there are spillover effects from default likelihood into stock returns, which in turn impact credit spreads. If credit spreads are indirectly influenced through stock returns, identifying this information channel would be particularly insightful.

The results show a drift in credit spreads for one lag followed by reversal in subsequent lags. This effect is clearly observed in table 5. Interestingly credit spreads are not influenced by lagged default likelihood. This in some way validates reduction of bias because LLMs ignore prior information in assessing transcripts of each meetings. The influence of credit spreads to default likelihood is contemporaneous only. Although market reaction to the information does take time observed in the drift. This is an indication of market friction due to diffusion of information.

In the case of stock prices, no significant effect from default likelihood is observed, indicating that stock values are not influenced by default likelihood. It is natural to assume that stock prices are driven more by discussions of growth rather than by default likelihood. As a result, spillover effects are not observed, since default likelihood does not impact stock prices. However, stock prices affect credit spreads and vice versa but that channel is not through the default likelihood.

4.4 Machine Learning Predictions

The drift observed in credit spreads in time series analysis and the lag effects from default likelihood could be linked to the time required to process and diffuse information due to variations in attention and interpretation. To test this hypothesis, one could create a counterfactual dataset where rational beliefs about unstructured data, combined with structured data such as ratings, firm fixed effects, liquidity-related measures, and duration, are used to predict credit spreads. The idea is that these predicted spreads would represent a scenario where two technologies coexist: one for rationally interpreting unstructured transcripts and another employing machine learning techniques to analyze features and determine optimal weights for prediction. In such a world, credit spreads would reflect all available information and absorb it immediately. Consequently, if all information were assimilated contemporaneously, there would be no auto-regression, meaning that past credit spreads should not influence current credit spreads.

To test this, I use Random Forest regression. The results are presented in Table 5. The Random Forest regression is performed on a full set of features, with the data divided into training and test sets, representing 67% and 33% of the sample, respectively. The training set uses the default parameters from Python's "Scikit-learn" package. I also optimized the parameters, but the results were not materially different. Figure 2 shows the importance of different features in the trained model. Although default likelihood contributes to prediction, it impact is smaller compared to obvious factors like ratings. The model is then applied to out-of-sample features, and the predicted credit spreads are compared to the actual out-of-sample credit spreads. Figure 1 displays the model fit for the Random Forest regression on the out-of-sample test data. The R-squared value for the predicted Random Forest regression is approximately 0.72. Random Forest regression is designed to reduce over-fitting so an R^2 of 0.72 is impressive.

Once the predicted credit spreads are generated, the next step is to test whether these predicted spreads reduce auto-regression by fitting auto-regression models on both the predicted and actual credit spreads. The results of this analysis are presented in Table 5. The actual credit spreads shown are residuals from regressions with firm fixed effects, which isolate the effects related to firm-specific aspects. The actual credit spreads exhibit clear auto-regressive coefficients up to lag 4, consistent with the time series analysis discussed earlier. There is a strong drift in the first lag with a coefficient of 0.5414 followed by reversal of lower magnitude -0.0369. In contrast, the predicted credit spreads, as shown in the table, exhibit no auto-regressive coefficients; all coefficients are insignificant. This indicates that predicted credit spreads are not influenced by their lags. This result demonstrates that technologies for rational belief formation through LLMs, combined with machine learning techniques, effectively reflect all available information in credit spreads, thereby making the market more complete.

An important question is whether the disappearance of auto-regression is due to power of machine learning or default likelihood measure also contributes to it To test it, I create an OLS predicted credit spreads. Instead of using all regressors, I use default likelihood as the only regressor. In this regression, I compare it to the observed credit spreads (not residuals from firm fixed effec). This is a parsimonous model that only uses one regressor. Then I fit the autoregression model of 3 lags.

The results are shown in Table 6. The columns represent autoreggression coefficients of observed and predicted credit spreads. A few notable observations in the results show that autoregression coefficients although signifiant in the predicted sample, have a much lower magnitude. The intercept contains most of the predicted value of 2.03% which is close to 2.61% of mean credit spreads of the sample. This shows that default likelihoods contributes in reducing PEAD in the corporate credit spreads. Second observation is that R^2 of the

observed AR model is 0.4705 while R^2 of the predicted observation is only 0.0206. This shows that lags have lower explanatory power in predicted versus observed sample.

From these tests, one can conclude that both machine learning and LLMs contribute predictive value and enhance market completeness. It is also clearly shown that LLM's interpreted default signals help reduce PEAD contributing to making markets efficient.

5 Conclusion

The paper examines whether textual information embedded in the conversation between management and analysts during earnings call transcripts provides a predictive signal, specifically regarding default likelihood. Since credit spreads reflect the market's implied default probability, this paper tests the impact of default likelihood signals on credit spreads. Extracting meaning from text is often fraught with issues like attention, bias, and interpretive dispersion among individuals processing the information. However, the use of large language models (LLMs) can mitigate bias and dispersion due to their ability to interpret textual meaning. LLMs are designed to capture the statistical properties of word meanings within sentences and can identify hidden meanings by considering the context of surrounding words. As a result, it is expected that LLM-interpreted signals will effectively predict credit spreads.

The issue of interpretation is closely related to the diffusion of information. The same frictions that cause interpretative dispersion and bias also extend the time required for information to be fully assimilated. This diffusion of information contributes to the drift observed in the auto-regressive coefficients of credit spreads. Therefore, LLMs may facilitate the identification of rational beliefs, potentially reducing the time needed for information diffusion and minimizing the associated drift.

The paper tests these hypotheses by feeding transcripts of conversations between management and analysts into ChatGPT (an LLM) and asking it to quantify the meaning of the transcript in terms of default likelihood. This default likelihood is then used as a regressor against credit spreads, alongside various controls, to determine if it adds predictive value. The results convincingly show that it does. Additionally, the paper employs machine learning predictions as counterfactual data to test whether predicted spreads incorporating default likelihood information reduce auto-regression. Further test with in a parsimonious model with default likelihood as the only predictor in OLS also shows that the magnitude of drift reduces considerably, strongly suggesting that LLMs contribute to market completeness. The LLM's signals effectively predict credit spreads and reduce drift, thus shortening the time required for information diffusion.

The paper paves the way for research that leverages machine learning to create counterfactual data for hypothesis testing. In the future, as these models become fully integrated into decision-making processes by market participants, it will be intriguing to observe whether drift in credit spreads is further reduced. This would provide additional confirmation that LLMs and machine learning are valuable tools in mitigating market frictions and enhancing market completeness.

Table 1: Summary Statistics

Summary Statistics of major variables for the paper. Default likelihood is assessed by ChatGPT, which rates the likelihood of a company defaulting on a scale of 1-10 based on summarized transcripts between the company and analysts. For each bond, credit spread is calculated on the last trade of the month for each bond. Then for each month, the credit spread represents the average of credit spreads of all bonds traded. The regression controls for bond maturity, measured as the number of days from the last trade date to expiration for each bond and then average maturity of all bonds for each company in the month. For the total dollar value of each bonds traded during is then added for all bonds for the company in the valuation month. Additional controls include S&P ratings (with partial ratings like BBB+, BBB, BBB-). Word count represents the total numbers of words spoken in the transcript of conversation in each earnings call. Market value of bond is the sum of bond's last trading price for each firm within the month. Notional bonds traded is the face value of total bonds outstanding for each firm in a month.

Variable	N	Mean	SD	Median
Default Likelihood	23,125	4	2	3
Word Count	23,125	8,502	2,342	8,563
Dollar Value Bonds Traded	23,125	$476.92	$1,381.93	$102.79
Maturity	23,125	2,886	1,622	2,490
Market Value Bond	23,120	$54.42	$36.21	$45.57
Market Cap	22,981	$38,483.17	$152,000.58	$10,546.62
Credit Spreads	22,307	2.6109	3.6198	1.6825
Notional Bonds Traded	23,125	$494.23	$1,766.79	$100.06
Average Stock Return	20,062	$0.00	$0.00	$0.00
Average Bid/Offer Spread	22,664	0.5276	0.5075	0.3949

Variable	P1	P25	P75	P99	Min	Max
Default Likelihood	1	2	5	9	0	10
Word Count	2,856	7,174	9,682	15,068	470	35,215
Dollar Value Bonds Traded	$0.11	$27.46	$354.44	$6,279.36	$0.00	$34,074.83
Maturity	397	1,734	3,846	7,945	-13	16,680
Market Value Bond	$6.04	$31.49	$66.60	$196.20	$0.00	$349.01
Market Cap	$193.99	$3,462.33	$30,997.24	$377,942.33	$18.05	$8,133,873.53
Credit Spreads	0.1472	0.9567	3.2132	14.8288	3e-04	99.8200
Notional Bonds Traded	$0.11	$26.82	$344.27	$6,385.41	$0.00	$77,185.50
Average Stock Return	$0.00	$0.00	$0.00	$0.00	$0.00	$0.00
Average Bid/Offer Spread	0.0400	0.2550	0.6300	2.5089	0.0000	13.4800

Table 2: Default Likelihood Affect on Credit Spreads

The regression analyzes the impact of default likelihood on credit spreads. Default likelihood is assessed by ChatGPT, which rates the likelihood of a company defaulting on a scale of 1-10 based on summarized transcripts between the company and analysts. For each bond, credit spread is calculated on the last trade of the month for each bond. Then for each month, the credit spread represents the average of credit spreads of all bonds traded. The regression controls for bond maturity, measured as the number of days from the last trade date to expiration for each bond and then average maturity of all bonds for each company in the month. For the total dollar value of each bonds traded during is then added for all bonds for the company in the valuation month. Additional controls include S&P ratings (with partial ratings like BBB+, BBB, BBB-) and firm fixed effects based on ticker symbols. Boom is an indicator variable that takes the value of 1 for observations between 2004 and 2007 and a 0 for other observations.

	CS Model	*Investment Grade*	*High Yield*	*Boom Model*
	Dependent variable:	*Dependent variable:*	*Dependent variable:*	*Dependent variable:*
	Credit Spreads	Credit Spreads	Credit Spreads	Credit Spreads
Default Likelihood	0.034***	0.047***	0.020***	−0.132
	(0.008)	(0.013)	(0.006)	(0.239)
Stock Return	0.662***	0.823***	−0.987***	0.036***
	(0.234)	(0.314)	(0.237)	(0.008)
Trade Spreads	−0.0002***	−0.0003***	−0.0001***	0.659***
	(0.00002)	(0.00004)	(0.00001)	(0.234)
Maturity	2.302***	3.236***	1.110***	−0.0002***
	(0.039)	(0.064)	(0.026)	(0.00002)
Boom				2.299***
				(0.039)
Boom X Default Likelihood				−0.087
				(0.059)
Ratings Fixed Effects	Yes	Yes	Yes	Yes
Firm Fixed Effects	Yes	Yes	Yes	Yes
Observations	18,790	11,402	7.388	18,790
R^2	0.563	0.543	0.483	0.563
Adjusted R^2	0.544	0.518	0.455	0.544
Residual Std. Error	2.335 (df = 17995)	2.825 (df = 10813)	1.007 (df = 7007)	2.335 (df = 17993)
F Statistic	29.190*** (df = 794; 17995)	21.844*** (df = 588; 10813)	17.219*** (df = 380; 7007)	29.153*** (df = 796; 17993)
Note:				*p<0.1; **p<0.05; ***p<0.01

Table 3: Analysis of Variance and Wald Test Results

The validity of the model is tested using Analysis of Variance (ANOVA) and Wald Test. Two models are tested. The full model contains all regressors in the table 2. The restricted model removes default likelihood from the regression. Default likelihood is assessed by ChatGPT, which rates the likelihood of a company defaulting on a scale of 1-10 based on summarized transcripts between the company and analysts. For each bond, credit spread is calculated on the last trade of the month for each bond. Then for each month, the credit spread represents the average of credit spreads of all bonds traded. The regression controls for bond maturity, measured as the number of days from the last trade date to expiration for each bond and then average maturity of all bonds for each company in the month. For the total dollar value of each bonds traded during is then added for all bonds for the company in the valuation month. Additional controls include S&P ratings (with partial ratings like BBB+, BBB, BBB-) and firm fixed effects based on ticker symbols.

Measure	Statistic	Value	P.Value
ANOVA	RSS	98229.3534 (NA) vs. 98138.8903 (1)	
	F	16.5876	0.0000
Wald Test	F	16.5876	0.0000

Table 4: Time Series Analysis (VAR)

The time series analysis using residuals of credit spreads, default likelihood, and stock prices. Each residual is calculated after regressing each of these variables on conrols used in table 2 and then calculating residuals from predicted values based on regression coeffiicients. Default likelihood is assessed by ChatGPT, which rates the likelihood of a company defaulting on a scale of 1-10 based on summarized transcripts between the company and analysts. For each bond, credit spread is calculated on the last trade of the month for each bond. Then for each month, the credit spread represents the average of credit spreads of all bonds traded. The regression controls for bond maturity, measured as the number of days from the last trade date to expiration for each bond and then average maturity of all bonds for each company in the month. For the total dollar value of each bonds traded during is then added for all bonds for the company in the valuation month. Additional controls include S&P ratings (with partial ratings like BBB+, BBB, BBB-)

Equation	Term	Estimate	Std. Error	t value	P-Value
Residual Credit Spread	Residual Credit Spread Lag 1	0.403***	0.01	49.07	0.00
Residual Credit Spread	Residual Default Likelihood Lag 1	-0.000	0.01	-0.01	1.00
Residual Credit Spread	Residual Stock Price Lag 1	0.002***	0.00	4.55	0.00
Residual Credit Spread	Residual Credit Spread Lag 2	-0.021*	0.01	-2.40	0.02
Residual Credit Spread	Residual Default Likelihood Lag 2	0.003	0.01	0.33	0.74
Residual Credit Spread	Residual Stock Price Lag 2	-0.002***	0.00	-3.30	0.00
Residual Credit Spread	Residual Credit Spread Lag 3	-0.058***	0.01	-7.02	0.00
Residual Credit Spread	Residual Default Likelihood Lag 3	0.011	0.01	1.39	0.16
Residual Credit Spread	Residual Stock Price Lag 3	0.001*	0.00	2.46	0.01
Residual Default Likelihood	Residual Credit Spread Lag 1	-0.006	0.01	-0.69	0.49
Residual Default Likelihood	Residual Default Likelihood Lag 1	0.024**	0.01	2.88	0.00
Residual Default Likelihood	Residual Stock Price Lag 1	-0.001	0.00	-1.52	0.13
Residual Default Likelihood	Residual Credit Spread Lag 2	0.011	0.01	1.19	0.23
Residual Default Likelihood	Residual Default Likelihood Lag 2	-0.013	0.01	-1.57	0.12
Residual Default Likelihood	Residual Stock Price Lag 2	0.000	0.00	0.12	0.91
Residual Default Likelihood	Residual Credit Spread Lag 3	-0.009	0.01	-1.03	0.30
Residual Default Likelihood	Residual Default Likelihood Lag 3	-0.028***	0.01	-3.46	0.00
Residual Default Likelihood	Residual Stock Price Lag 3	0.000	0.00	0.52	0.61
Residual Stock Price	Residual Credit Spread Lag 1	0.970***	0.17	5.76	0.00
Residual Stock Price	Residual Default Likelihood Lag 1	0.299	0.16	1.86	0.06
Residual Stock Price	Residual Stock Price Lag 1	0.754***	0.01	91.64	0.00
Residual Stock Price	Residual Credit Spread Lag 2	-0.504**	0.18	-2.77	0.01
Residual Stock Price	Residual Default Likelihood Lag 2	0.078	0.16	0.48	0.63
Residual Stock Price	Residual Stock Price Lag 2	-0.055***	0.01	-5.37	0.00
Residual Stock Price	Residual Credit Spread Lag 3	0.062	0.17	0.37	0.71
Residual Stock Price	Residual Default Likelihood Lag 3	0.116	0.16	0.72	0.47
Residual Stock Price	Residual Stock Price Lag 3	0.028***	0.01	3.42	0.00

Table 5: Auto Regressive Model of Credit Spreads

Autoregressive models on the residuals of actual credit spreads versus predicted by machine learning Random Forest model. The residual of actual credit spreads are calculated after regressing credit spreads with the firm fixed effects. The Random Forest regression uses the features that are regressors in table2 in predicting credit spreads. Default likelihood is assessed by ChatGPT, which rates the likelihood of a company defaulting on a scale of 1-10 based on summarized transcripts between the company and analysts. For each bond, credit spread is calculated on the last trade of the month for each bond. Then for each month, the credit spread represents the average of credit spreads of all bonds traded. The regression controls for bond maturity, measured as the number of days from the last trade date to expiration for each bond and then average maturity of all bonds for each company in the month. For the total dollar value of each bonds traded during is then added for all bonds for the company in the valuation month. Additional controls include S&P ratings (with partial ratings like BBB+, BBB, BBB-)

Statistic	**Actual**	**Predicted With Default Likelihood**
	Value	**Value**
No. Observations	14824	4892
Log Likelihood	-32328.411	-11753.467
S.D. of Innovations	2.144	2.681
AIC	64670.823	23520.933
BIC	64724.048	23566.393
HQIC	64688.493	23536.885

Variable	**Actual**			**Predicted With Default Likelihood**		
	coef	**z**	**P> |z|**	**coef**	**z**	**P> |z|**
const	0.0017	0.097	0.923	2.5847***	28.826	0.000
y.L1	0.5144***	62.658	0.000	-0.0205	-1.433	0.152
y.L2	-0.0369***	-3.996	0.000	0.0030	0.208	0.835
y.L3	-0.0420***	-4.551	0.000	-0.0073	-0.513	0.608
y.L4	-0.0285**	-3.088	0.002	-0.0209	-1.460	0.144
y.L5	-0.0373***	-4.545	0.000	0.0031	0.213	0.831

Table 6: Autoregression (3) for Actual vs Predicted Credit Spreads

The autoregression (3) models created with observed credit spreads and predicted credit spreads. Observed credit spreads are calculated by taking the difference between corporate bond's yield based on last traded price each month and subtracting closest constant maturity Tresasury Yield. Predicted credit spreads are based on spreads prediced using credit spreads using default likelihood. Default likelihood is assessed by ChatGPT, which rates the likelihood of a company defaulting on a scale of 1-10 based on summarized transcripts between the company and analysts.

Term	Coefficient (Observed Credit Spreads)	T-stat (Actual)	Coefficient (Predicted Credit Spreads)	T-stat (DL)
(Intercept)	0.7386***	31.4689	2.037***	74.6055
Lag 1	0.6214***	91.0646	0.1003***	14.9997
Lag 2	0.1103***	13.463	0.0725***	10.8195
Lag 3	-0.0112	-1.599	0.0469***	7.0148
R^2	0.4795		0.0208	
Adjusted R^2	0.4795		0.0206	
Observations	21533		22304	

Figure 1: Prediction With Default Likelihood

The Random Forest regression uses the features that are regressors in table2 in predicting credit spreads. The training is done on two third sample and the test is on one-third of the remaining sample. The graph represents the fit between predicted credit spreads on the test sample and the actual credit spread in the same test sample

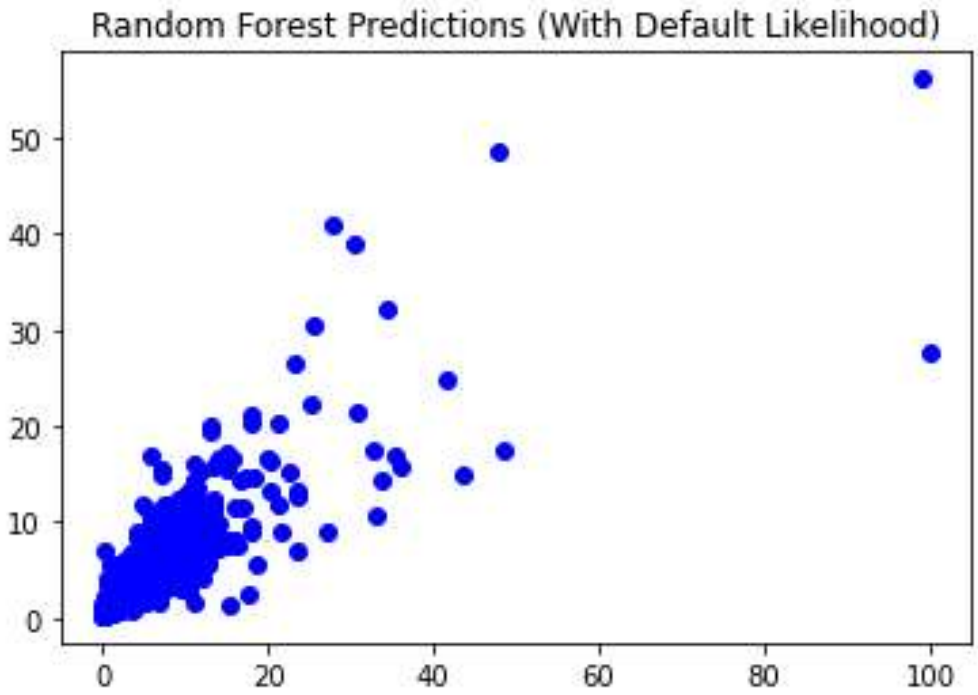
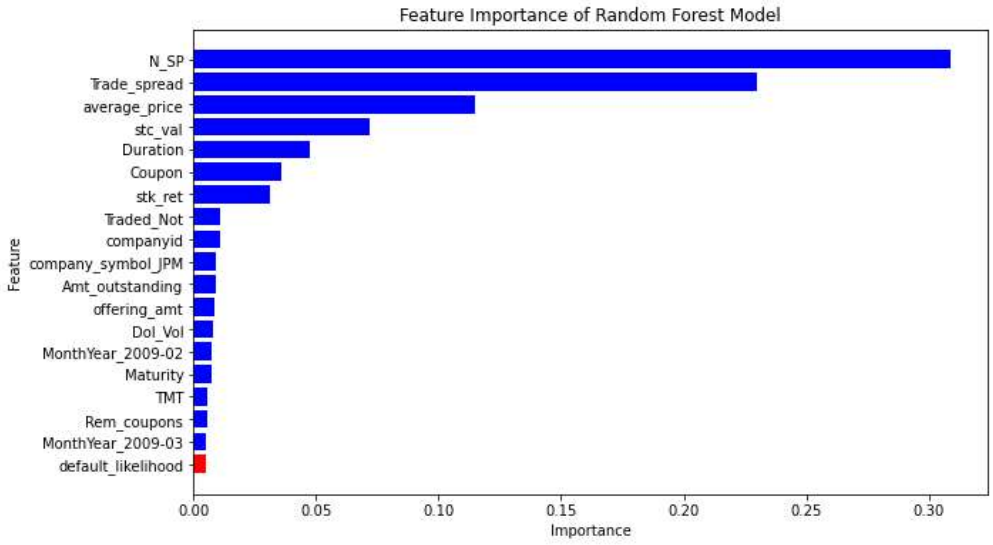


Figure 2: Feature Importance With Default Likelihood

The Random Forest regression uses the features that are regressors in table2 in predicting credit spreads. The training is done on two third sample and the test is on one-third of the remaining sample. The graph represents the feature importance from the training set. It contains all features that are shows better fit than default likelihood. All features that shows better fit than default likelihood are added together as "other"



References

Angeletos, G.-M., Huo, Z., and Sastry, K. A. (2021). Imperfect macroeconomic expectations: Evidence and theory. *NBER Macroeconomics Annual*, 35(1):1–86.

Baker, M. and Wurgler, J. (2007). Investor sentiment in the stock market. *Journal of economic perspectives*, 21(2):129–151.

Barberis, N., Shleifer, A., and Vishny, R. (1998). A model of investor sentiment. *Journal of financial economics*, 49(3):307–343.

Bernard, D., Blankespoor, E., de Kok, T., and Toynbee, S. (2023). A modular measure of business complexity. *Available at SSRN 4480309*.

Bernard, V. L. and Thomas, J. K. (1989). Post-earnings-announcement drift: delayed price response or risk premium? *Journal of Accounting research*, 27:1–36.

Brants, T., Popat, A., Xu, P., Och, F. J., and Dean, J. (2007). Large language models in machine translation. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 858–867.

Bybee, J. L. (2023). The ghost in the machine: Generating beliefs with large language models. *arXiv preprint arXiv:2305.02823*.

Bybee, L., Kelly, B., and Su, Y. (2023). Narrative asset pricing: Interpretable systematic risk factors from news text. *The Review of Financial Studies*, 36(12):4759–4787.

Chan-Lau, J. A. (2006). Market-based estimation of default probabilities and its application to financial market surveillance. IMF Working Paper WP/06/104, International Monetary Fund.

Chen, Y., Andiappan, M., Jenkin, T., and Ovchinnikov, A. (2023). A manager and an ai walk into a bar: does chatgpt make biased decisions like we do? *Available at SSRN 4380365*.

Chen, Y., Kelly, B. T., and Xiu, D. (2022). Expected returns and large language models. *Available at SSRN 4416687*.

Coibion, O. and Gorodnichenko, Y. (2012). What can survey forecasts tell us about information rigidities? *Journal of Political Economy*, 120(1):116–159.

Daniel, K., Hirshleifer, D., and Subrahmanyam, A. (1998). Investor psychology and security market under-and overreactions. *the Journal of Finance*, 53(6):1839–1885.

Donovan, J., Jennings, J., Koharki, K., and Lee, J. (2021). Measuring credit risk using qualitative disclosure. *Review of Accounting Studies*, 26:815–863.

Eisfeldt, A. L., Schubert, G., and Zhang, M. B. (2023). Generative ai and firm values. Technical report, National Bureau of Economic Research.

Gebhardt, W. R., Hvidkjaer, S., and Swaminathan, B. (2005). Stock and bond market interaction: Does momentum spill over? *Journal of Financial Economics*, 75(3):651–690.

Greenwood, R. and Hanson, S. G. (2013). Issuer quality and corporate bond returns. *The Review of Financial Studies*, 26(6):1483–1525.

Greenwood, R., Hanson, S. G., and Jin, L. J. (2019). Reflexivity in credit markets. Technical report, National Bureau of Economic Research.

Hong, H. and Stein, J. C. (1999). A unified theory of underreaction, momentum trading, and overreaction in asset markets. *The Journal of finance*, 54(6):2143–2184.

Jegadeesh, N. and Titman, S. (1993). Returns to buying winners and selling losers: Implications for stock market efficiency. *The Journal of finance*, 48(1):65–91.

Jegadeesh, N. and Titman, S. (1995). Overreaction, delayed reaction, and contrarian profits. *The Review of financial studies*, 8(4):973–993.

Jha, M., Qian, J., Weber, M., and Yang, B. (2024). Chatgpt and corporate policies. Technical report, National Bureau of Economic Research.

Kim, A., Muhn, M., and Nikolaev, V. (2023a). Bloated disclosures: Can chatgpt help investors process financial information? *arXiv preprint arXiv:2306.10224*.

Kim, A., Muhn, M., and Nikolaev, V. (2023b). From transcripts to insights: Uncovering corporate risks using generative ai. *arXiv preprint arXiv:2310.17721*.

Kim, A., Muhn, M., and Nikolaev, V. V. (2024). Financial statement analysis with large language models. *Chicago Booth Research Paper Forthcoming, Fama-Miller Working Paper*.

Kriebel, J. and Stitz, L. (2022). Credit default prediction from user-generated text in peer-to-peer lending using deep learning. *European Journal of Operational Research*, 302(1):309–323.

Manela, A. and Moreira, A. (2017). News implied volatility and disaster concerns. *Journal of Financial Economics*, 123(1):137–162.

Siddiqui, H. U. R., de Abajo, B. S., de la Torre Díez, I., Rustam, F., Raza, A., Atta, S., and Ashraf, I. (2023). Predicting bankruptcy of firms using earnings call data and transfer learning. *PeerJ Computer Science*, 9:e1134.

Van Binsbergen, J. H., Han, X., and Lopez-Lira, A. (2023). Man versus machine learning: The term structure of earnings expectations and conditional biases. *The Review of financial studies*, 36(6):2361–2396.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30.

Wu, Z., Dong, Y., Li, Y., and Shi, B. (2023). Unleashing the power of text for credit default prediction: Comparing human-generated and ai-generated texts. *Available at SSRN 4601317*.