ChatGPT

Financial PhraseBank and Financial Sentiment Datasets (2019–2024)

Financial PhraseBank Dataset Overview

The **Financial PhraseBank** is a widely used benchmark dataset for financial sentiment analysis. It consists of roughly **4,840 English sentences** (mostly news headlines or short statements) about companies, drawn from financial news articles and press releases [1] [2]. Each sentence is labeled with one of three sentiment classes – **positive**, **negative**, or **neutral** – representing the sentence's sentiment **from the perspective of an investor** [3]. In other words, annotators labeled whether the news in the sentence would likely have a positive, negative, or neutral impact on the company's stock price [4]. Sentences with no clear financial impact or irrelevant sentiment are labeled *neutral* by this definition [5].

Annotation methodology: The dataset was created by Malo et al. (2014) to provide high-quality training data for financial sentiment models [6]. Sixteen annotators with finance/economics backgrounds (3 researchers and 13 MSc students) each tagged a subset of sentences [7] [8]. Each sentence received **5–8 independent annotations**, allowing measurement of inter-annotator agreement [9]. Based on these overlapping annotations, the creators released **four variants** of the PhraseBank with different agreement thresholds [10] [11]:

- *50% agreement (Full set):* 4,840 sentences (at least half the annotators agreed on the label) [12]. This is the largest set, often used as the full dataset.
- *66% agreement:* 4,217 sentences (two-thirds of annotators agreed) [12].
- *75% agreement:* 3,453 sentences (¾ of annotators agreed) [12].
- *100% agreement:* 2,264 sentences (all annotators agreed on the sentiment) [13].

These subsets let researchers trade off **data quantity vs. label certainty**. The 100%-agreement subset has fewer examples but very reliable labels (unambiguous sentiment), whereas the 50% set includes more data at the risk of some label noise. Notably, even in the full set, many sentences have high agreement – e.g. ~70% of sentences have ≥66% annotator agreement [11].

Dataset structure: Financial PhraseBank is typically provided as a two-column text corpus (often a CSV): one column for the sentiment label (Negative, Neutral, Positive) and one for the sentence (news headline or phrase) [2]. An example entry would be: *"neutral" – "The company has no plans to cut its dividend."* The sentences range up to a few hundred characters (max ~300 tokens [14]) since they are often headlines or single-sentence news summaries.

Use cases: Since 2019, the PhraseBank has become a **standard benchmark** for financial sentiment classification. It is used to **train and evaluate** models that can read financial news and predict sentiment, which is valuable for:

- **Investor information systems:** e.g. highlighting positive vs negative news to traders and analysts [15].
- **Quantitative trading strategies:** using news sentiment signals (extracted via these models) to predict stock movements or volatility.

- **Transfer learning in domain-specific NLP:** PhraseBank serves as a fine-tuning target for language models adapted to finance (as discussed later with FinBERT, etc).
- **Benchmarks in research:** New models or methods in financial NLP often report their accuracy on PhraseBank to demonstrate state-of-the-art performance [16] [17]. It's also combined with other datasets (like FiQA, see below) for broader evaluation [18].

Overall, the Financial PhraseBank provides a well-defined, human-annotated standard for **sentence-level sentiment in financial news**, which has underpinned much of the progress in financial text analysis in the last five years.

Exploratory Data Analysis (EDA) and Dataset Characteristics

Researchers often begin with **exploratory data analysis (EDA)** on Financial PhraseBank (and similar datasets) to understand class distributions and linguistic patterns. Key observations and techniques include:

- **Class Imbalance:** The PhraseBank is highly skewed toward the **neutral** class. Roughly *60% of the sentences are neutral*, with the remaining 40% split between positive and negative [19]. In all released subsets, neutral dominates (~59–62% of examples), while positives are about 25–28% and negatives only ~12–13% [20]. For example, one analysis of the full 50% agreement set found ~59.4% neutral, ~28.2% positive, and ~12.5% negative sentences [21]. This imbalance reflects that many news statements are factual or mixed in tone (hence neutral). **Implication:** Models can get biased toward predicting neutral, so evaluating with metrics like macro-F1 (which weights each class equally) is important [19].

- **Label Distribution & Agreement:** As expected, the smaller high-agreement subsets have slightly different distributions (e.g. the 100%-agreement data had ~61% neutral [21]). Unanimously labeled sentences tend to be more clearly positive or negative events, but interestingly a majority still came out neutral, indicating many news items unanimously have no obvious positive/negative impact. This highlights the prevalence of neutral, factual content in financial news.

- **Common Words and Phrases:** EDA on keyword frequency shows that **certain words strongly correlate with sentiment** in this domain. For instance, words about growth and profit (e.g. "increased", "rise", "growth", "profit", "exceed") frequently appear in positive sentences, whereas words indicating decline or loss (e.g. "falling", "losses", "drop", "decline") are common in negatives. Domain-specific terms also matter: for example *"liability"* or *"impairment"* are negative in a financial context, while *"accretive"* or *"upgrade"* are positive. Researchers often cross-reference **financial sentiment lexicons** (like Loughran–McDonald 2011) during EDA to identify such domain-specific sentiment cues [22]. A lexicon-based tally on PhraseBank data will typically find far more occurrences of negative words than positive (financial texts tend to contain many risk-related terms), but their actual sentiment depends on context [23]. Modern models thus need to understand context beyond just these keywords.

- **Sentence Length and Structure:** Another EDA point is the length distribution of sentences. PhraseBank sentences are relatively short (often headlines). One study noted an average length well under 100 tokens, with the longest ~80 tokens [24]. Many headlines are concise and omit pronouns or context, which can make sentiment interpretation challenging. E.g., *"Profit up 5% on higher revenue, but costs weigh on outlook"* – a short headline containing mixed sentiments. Understanding **nuanced sentence structure** (e.g. contrasting clauses, negations like "despite", "however") is crucial. EDA might involve parsing a few examples to see how positives and negatives can co-occur in one sentence [25].

- **Class-Specific Keyword Analysis:** Some works perform **word cloud or frequency analysis per class**. While not always published in papers, open-source explorations show, for example, that **negative sentences often contain numeric declines** (e.g. "fell *X*%", "loss of *Y* million"), whereas **positive sentences feature gains** ("grew *X*%", "record profit of ..."). Neutral sentences often report announcements or facts without strong evaluative language ("Company issued *new shares*", "CEO *resigned*" – which might not be intrinsically positive or negative without context). Identifying such patterns helps in feature engineering for traditional models or for interpreting model behavior.

In summary, EDA confirms that **neutral instances dominate** financial sentiment data and that **surface keywords can be misleading** without context (a classic example: "surged" is positive in "shares surged 5%", but negative in "costs surged 5%"). These insights have driven researchers to adopt appropriate data handling (e.g. resampling or weighted loss for imbalance [26]) and to prefer context-sensitive models over simple keyword spotting.

Preprocessing and Text Preparation Pipelines

Preprocessing financial text data is a crucial step, varying slightly between traditional machine learning pipelines and modern transformer-based approaches:

- **Text Normalization:** Standard NLP cleaning steps are applied, such as lowercasing (for case-insensitive models), removing HTML tags or irrelevant symbols, and often **preserving financial indicators**. For example, stock tickers (e.g. `NYSE: JPM`) or currency symbols may appear in news; rather than stripping them, they are often left in or replaced with descriptive tokens (like `<TICKER>` or `<CUR>`), since they carry information. Numerical figures (percentages, monetary values) might be normalized (replaced with a generic `<NUM>` token) in some pipelines to reduce sparsity, though transformers can handle numbers as is by subword tokenization. In the PhraseBank, many sentences contain numbers and percentages; most recent works keep them as-is because models like BERT can ingest them (often split into digits tokens) and sometimes infer their magnitude contextually [24].

- **Tokenization Strategies:** Financial text may include domain-specific jargon and multi-word expressions ("credit default swap", "earnings per share"). Tokenization must handle these gracefully. Modern approaches use **subword tokenization** (BERT's WordPiece or RoBERTa's Byte-Pair Encoding) which can split uncommon financial terms into meaningful pieces. For example, "EBITDA" might be tokenized as "EBIT" + "DA" or as a single token if in the vocabulary of a domain model. Pre-BERT, some studies used custom tokenizers or kept financial abbreviations intact. Today, using the pretrained model's tokenizer is standard (FinBERT and others come with vocabularies that include many financial terms). If using classic methods (like TF-IDF or word embeddings), one might apply **stemming/lemmatization** to reduce inflection (e.g. "rises", "rising" → "rise") and remove stopwords, but **note:** Many common stopwords (like "not", "without") can flip sentiment, so financial sentiment pipelines often **retain negation words** to not lose meaning [25].

- **Use of Financial Lexicons:** Incorporating domain knowledge via lexicons is a common preprocessing or feature engineering step, especially in earlier works. The **Loughran-McDonald financial sentiment lexicon** [27], which contains word lists tailored to financial contexts (e.g.

words like “liability” or “collateral” marked as negative, which general lexicons mislabel), has been used to **augment features**. For example, a simple approach is adding features like “# of negative words from L&M lexicon in the sentence” or “net sentiment score from lexicon” as inputs to a classifier. Some research (pre-2018) relied on lexicon-based classification as a baseline: e.g. counting positive vs negative words in a headline [23]. However, pure lexicon methods often misclassify sentences where context matters (e.g., “**no** decline in losses” has two negative words *no* and *losses* but the overall sentiment is positive because losses didn’t increase). Thus, recent pipelines use lexicons more subtly – for instance, to **initialize word embeddings** or as an auxiliary input to neural models. One modern example is **FinSSLx** (Maia et al. 2018), which performed a “sentiment lexicon simplification” step: breaking complex sentences into shorter units and using a lexicon to help classify each piece [28].

- **Handling Out-of-Vocabulary (OOV) Terms:** Financial news introduces new company names, product names, or abbreviations regularly. Transformer tokenizers mitigate this via subwords, but older models had OOV issues. A preprocessing step in some projects is to **replace company names or proper nouns with a generic placeholder**, so the model focuses on the sentiment context rather than the specific entity. For example, “Tesla’s profit rises” → “[COMPANY]’s profit rises”. This can help if a company name would otherwise be an OOV token or overly specific. Similarly, rare financial acronyms may be expanded or explained in a preprocessing step if not recognized by the model.

- **Data Augmentation:** In the past five years, data augmentation for text has been explored to address the relatively small size of labeled financial datasets. Common NLP augmentations like **back-translation** (translating a sentence to another language and back to English to get a rephrased version) or **synonym replacement** have been tested in general sentiment tasks [29]. In finance, augmentation is tricky due to jargon – replacing a word with a synonym might change nuance (e.g., “gain” vs “win” are not interchangeable in a financial context). A notable emerging approach is **using large language models (LLMs) to generate synthetic data**. Recent work has demonstrated that prompting GPT-3 or GPT-4 to **create additional financial sentences** of a given sentiment can improve model training [30]. For instance, researchers generated new positive and negative examples using GPT-4 (while preserving realism and context) and appended these to the training set [30]. A 2024 study reported that augmenting Financial PhraseBank with *GPT-4 generated sentences* led to higher F1-scores for BERT-based classifiers [31]. Another form of augmentation in Lopez Roldan (2024) was *instructional augmentation*, where the task instructions given to human annotators were used to produce paraphrased inputs for the model (essentially broadening the ways a sentiment question might be asked to an LLM) [32]. While not “augmentation” of the raw data per se, this falls under enriching the training signal. Overall, **synthetic data generation** has become an important technique to overcome the scarcity of labeled financial text, supplementing the ~5k real sentences with thousands of machine-crafted ones in some experiments [33].

In summary, modern preprocessing for financial sentiment focuses on **keeping important domain information** (numbers, tickers, negations) intact, possibly leveraging **domain lexicons**, and carefully expanding the dataset via **augmentation**. With transformers, heavy text cleaning is less needed (since the model can learn nuance from context), but handling class imbalance and limited data via augmentation or smart resampling remains key.

Modeling Approaches (2019–2024)

Approaches to financial sentiment classification have evolved from traditional machine learning to specialized fine-tuned transformers. Below we detail the spectrum of models and methodologies used, including architecture choices, fine-tuning strategies, and typical hyperparameters.

Traditional and Deep Learning Models (Pre-Transformer)

Before transformer models became dominant (pre-2018 and early 2019), researchers applied both classic algorithms and basic neural networks:

- **Lexicon & Rule-Based Models:** The earliest solutions often relied on dictionaries of positive/negative words. For example, Malo et al. (2014) in introducing PhraseBank developed a linear model incorporating phrase-structure features and a domain lexicon (their LPS – *Linearized Phrase Structure* model) [34]. Such lexicon-based classifiers achieved around 70–72% accuracy on the dataset [35]. These are lightweight but struggle with context (they might mislabel sentences that have negations or domain-specific usage). They were quickly surpassed by data-driven methods but remain a useful baseline.

- **Support Vector Machines and Others:** Many mid-2010s works treated sentiment as a standard text classification task using SVMs, Naive Bayes, or logistic regression with features like TF-IDF vectors or word embeddings [36] [37]. For instance, an SVM with bag-of-words on PhraseBank might reach ~70% accuracy, but was limited by the small dataset size and vocabulary mismatch issues.

- **Recurrent Neural Networks:** The introduction of deep learning saw **LSTM networks** applied to financial text. An LSTM can capture sequence information and short-term context, which is valuable for sentences with multiple clauses. Early attempts like Zhang et al. (2018) and others used LSTMs (sometimes with pretrained general embeddings like GloVe) on financial news. An LSTM baseline without any pretrained language model achieved about **71% accuracy (F1≈0.64)** on Financial PhraseBank [35] – essentially on par with the better classical methods. Improvements came from using richer embeddings: e.g. an **LSTM + ELMo embeddings** (contextual word embeddings from a language model) raised performance to ~75% accuracy (F1≈0.70) [38]. These results, from Araci (2019), show that *contextualized embeddings* already gave LSTMs a boost on this task. Another approach was **CNNs or hybrid models** – a 2018 paper by Krishnamoorthy used a hierarchical structure (HSC) combining word and sentence-level modeling, reaching about 76% F1 [39]. Overall, by 2018 the best non-transformer models (using advanced architectures or semi-supervised tricks) achieved **75–80% accuracy** on PhraseBank [38].

- **ULMFiT and Transfer Learning:** In 2018, ULMFiT (Universal Language Model Fine-tuning) introduced the idea of pretraining an LSTM on generic text then fine-tuning on target data. This method was applied to financial sentiment as well. Araci (2019) reports a **ULMFiT model** reached **83% accuracy, F1≈0.79** on PhraseBank [35] – a substantial gain over vanilla LSTMs. ULMFiT's success foreshadowed the even larger gains from transformers by leveraging unlabeled text to learn language patterns.

Hyperparameters (traditional models): These varied, but common choices were unigram or bigram features for SVM/LogReg with regularization tuned on a small validation set. For LSTMs, a hidden size of ~100–150, and pre-trained 300-dimensional embeddings (GloVe or Word2Vec trained on financial news) were typical. Training epochs were in the tens (due to small data size, models could train quickly). One study set an LSTM with 128 hidden units (bidirectional) and used **max-pooling** over time to capture the sentence representation [40]. Dropout ~0.5 was used to prevent overfitting on such a small dataset. These models often required careful early stopping because with only ~4k examples, they could overfit in a few epochs.

Transformer-Based Models and Fine-Tuned Language Models

The real breakthrough in financial sentiment analysis came with transformer language models and fine-tuning:

- **BERT Fine-Tuning:** Using BERT (Bidirectional Encoder Representations from Transformers) fine-tuned on the PhraseBank was a game-changer. Even the **general "Vanilla" BERT** (pretrained on Wikipedia/BooksCorpus) when fine-tuned for classification on PhraseBank achieved about **85–86% accuracy (macro-F1 ≈ 0.84)** [41] [42] – a huge jump over prior methods. This was first demonstrated around 2019. The fine-tuning process typically involves adding a classification layer on [CLS] token and training for a few epochs. Common hyperparameters for BERT fine-tuning on this dataset: *learning rate* 2e-5 to 5e-5 (AdamW optimizer), *batch size* ~16–32, *epochs* ~3–5. Because the dataset is small, many implementations use **cross-validation** to reliably evaluate BERT. For example, Araci (2019) used 10-fold cross-validation and found BERT's macro-F1 ~0.84 [41]. An important detail is handling class imbalance: some fine-tuning runs use a **weighted loss** (inverse-frequency class weights) to ensure the model learns the minority classes [19]. This helps maximize macro-F1 rather than just accuracy.

- **FinBERT (Domain-Specific BERT):** In 2019, researchers introduced **FinBERT**, a BERT model further **pre-trained on financial corpora**, to inject domain knowledge. One prominent FinBERT by Araci (2019) was pretrained on a large set of financial news (the Reuters TRC2 corpus with ~1.8M news articles) and corporate filings, before fine-tuning on PhraseBank [43]. The result was state-of-the-art: FinBERT reached **86% accuracy, F1 ≈ 0.84** on the full PhraseBank, roughly matching vanilla BERT on the full set but **shining on the high-agreement subset** – e.g. on the 100% agreement data FinBERT achieved **97% accuracy (F1 ≈ 0.95)** [44] [45]. This indicates FinBERT learned the domain so well that on unambiguous cases it was nearly perfect. FinBERT also significantly outperformed earlier models on **FiQA** (a related financial sentiment dataset – details below), proving the value of domain pre-training [46] [47]. The term "FinBERT" has since been used by multiple teams (Prosus AI's FinBERT, Huang et al.'s FinBERT, etc.), generally referring to BERT-base architectures with financial domain pre-training. These models are available open-source (e.g. **ProsusAI/finbert** on HuggingFace) and have become a go-to for financial NLP tasks [48].

- **RoBERTa and Other Transformers:** Following BERT, variants like **RoBERTa**, **DistilBERT**, and **ALBERT** were tried for financial sentiment. RoBERTa (which is similar architecture but trained longer on more data) often performs on par or slightly better than BERT. Some projects reported RoBERTa-base fine-tuned on PhraseBank achieving ~88–90% accuracy (no large published gap over BERT, but modest gains due to RoBERTa's stronger pretraining). **DistilBERT** (a lighter 66M-parameter model) was fine-tuned by practitioners as a speed/size trade-off; one GitHub report shows DistilBERT reaching ~82% accuracy (F1 ~0.81) on PhraseBank [49] – a bit lower than full BERT, but still far above earlier baselines. Newer transformer models like **DeBERTa (Decoding-enhanced BERT)** and **FinGPT (Financial GPT)** have also been adapted: for example, a HuggingFace user fine-tuned DeBERTa-v3 on a combined financial news dataset and achieved high performance (though exact numbers aren't formally published) [50]. In practice, any modern transformer with sufficient capacity can do well, but **domain-specific pretraining** consistently boosts performance by a few points.

- **FinancialBERT and Enhanced Models:** Beyond FinBERT, there are other specialized models. **FinancialBERT** (by Rachid, 2022) is a BERT-base model pretrained on a large financial text corpus (including news and filings) [51] . When fine-tuned on PhraseBank, it reportedly *"outperforms general BERT and other domain-specific models"* [52] . In fact, the published fine-tuning results for FinancialBERT are remarkable: a **weighted F1 of 0.98** on a test split of PhraseBank [53] , implying accuracy ~98%. This nearly perfect result suggests that with extensive pretraining and careful tuning, the model almost saturates the dataset (indeed, their precision/recall for neutral and positive classes are ~98–99% [54]). Such performance might partly reflect the simplicity of many examples and possibly the use of the easier agreement subset or a favorable train/test split. Nonetheless, it represents the **state-of-the-art on PhraseBank** – effectively, transformer models can now solve most PhraseBank instances correctly. Another 2023 study introduced **EnhancedFinSentiBERT**, which further incorporated financial domain knowledge into BERT and achieved top accuracy on PhraseBank [55] (exact numbers not cited in open text, but presumably on par with the above). The continuous improvement of in-domain models has narrowed the error margin: errors now tend to occur only on highly ambiguous or context-dependent sentences.

- **Sequence Length and Context:** Transformers can in theory handle longer inputs (BERT up to 512 tokens). While PhraseBank sentences are short, some research experimented with concatenating multiple sentences to provide additional context (termed *"financial phrasebank concatenate"* in one study) [56] . For example, concatenating a positive sentence with a negative one to simulate a longer text and training models on that. This is more an experimental setup for robustness (and related to Next Sentence Prediction tasks) [57] . Most sentiment models, however, treat each sentence independently (since that's how the dataset is structured), and a single sentence usually fits well within model length limits (often under 128 tokens). Thus, typical fine-tuning uses max_seq_length 128 or 256 (some choose 512 to be safe, as seen in FinancialBERT using 512 tokens [58]).

- **Fine-Tuning Strategies:** Fine-tuning can be sensitive with small data. Researchers have employed tricks like **discriminative learning rates** and **layer freezing**. For instance, FinBERT (Araci 2019) used a strategy inspired by ULMFiT: initially freeze most of BERT's layers and gradually unfreeze them during training, while using a slightly lower learning rate for lower layers [59] . This was done to avoid *catastrophic forgetting* of language understanding in early epochs. They also used a *slanted triangular learning rate schedule* (warm-up then decay) and even experimented with further pre-training BERT on the task data itself ("FinBERT-task") [60] . The differences were minor but the approach ensured stable training. In practice, many open implementations find that simply fine-tuning all layers of BERT with a low LR (2e-5) for ~3 epochs is sufficient to get >85% accuracy. One just has to monitor a validation split to avoid overfitting – with 3–5 epochs being the sweet spot (FinancialBERT chose 5 epochs at 2e-5 [61] ; another experiment found 3 epochs at 5e-5 worked well [62] [63]).

- **Hyperparameters (transformers):** Summarizing typical values – *optimizer:* AdamW; *learning rate:* 2e-5 (with linear warmup of ~10% of training steps) [64] ; *batch size:* 16 or 32; *epochs:* 3–5 (with early stopping if no improvement); *max length:* 128 or 256 (since all sentences <300 tokens [24]); *dropout:* 0.1 (the default in BERT) [65] . Some studies also tune the number of epochs via cross-val – e.g. one might find 4 epochs gives best validation F1. Fine-tuning often uses **macro-averaged F1** as the selection metric (to ensure good performance on minority classes) [19] .

The table below highlights performance of various models on the Financial PhraseBank (full dataset) as reported in the literature:

Model	**Accuracy**	**Macro F1**	**Notes**
Lexicon+Linear (Malo et al. 2014)	~71% [35]	0.71 [39]	“LPS” phrase-structure model with lexicons.
Hybrid CNN/LSTM (HSC, 2018)	~71% [39]	0.76 [39]	Hierarchical CNN for news (Krishnamoorthy 2018).
LSTM (baseline, 2019)	71% [35]	0.64 [35]	Bi-LSTM with glove embeddings (Araci 2019).
LSTM + ELMo (2019)	75% [38]	0.70 [38]	LSTM with contextual word embeddings.
ULMFiT (2019)	83% [42]	0.79 [42]	Pretrained LSTM LM fine-tuned (Araci 2019).
BERT-base (2019)	~85% [41]	0.84 [41]	Fine-tuned BERT, 10-fold CV (Araci 2019).
FinBERT (Araci 2019)	86% [42]	0.84 [42]	BERT with financial pretraining (Araci thesis).
BERT-base (student GitHub 2021)	86% [62]	0.86 [62]	3 epochs, LR 5e-5 on FPB [62] .
DistilBERT (student GitHub 2021)	82% [49]	0.81 [49]	Lighter model, fine-tuned on FPB [49] .
FinBERT (student GitHub 2021)	90.9% [66]	0.91 [66]	FinBERT fine-tuned 3 epochs, LR 5e-5 [66] .
FinancialBERT (Rachid 2022)	98% [54]	0.98 [54]	BERT on large fin. corpus, fine-tuned 5 epochs.

<small>**Table:** Performance of various models on Financial PhraseBank. (Accuracy and F1 for full 50%-agreement set unless otherwise noted.) FinBERT (2019) improved the state-of-art by ~15 points over prior models [67] [42] . By 2022, FinancialBERT nearly saturates the dataset with 98% accuracy in one report [54] .</small>

As seen, **transformer-based models (especially with domain adaptation) dominate performance**, making financial sentiment classification a largely solved task on this dataset. The remaining challenge is usually generalization: ensuring the model trained on PhraseBank transfers to other contexts (e.g. different time periods, different kinds of text like social media) – which leads to the next section.

Benchmark Results on Similar Datasets

To gauge model generality, researchers test on other financial sentiment datasets alongside PhraseBank. Two prominent English-language datasets from the last five years are **FiQA Sentiment** and the **Financial News Headline datasets** (e.g. from SemEval competitions or proprietary sources). We compare and summarize benchmarks on these:

- **FiQA Task 1 (2018):** This was introduced in the Financial QA/Sentiment challenge at WWW 2018 [68]. FiQA Task 1 is an **aspect-based financial sentiment dataset**. It contains **1,174 text snippets (news headlines and social media posts)** each paired with a target entity (company or financial asset) and a **sentiment score** [69]. The sentiment is a *continuous value* (not just class labels) representing crowd-annotated sentiment intensity towards the target. For example, a news headline "XYZ Corp shares hit record high on strong results" targeting *XYZ Corp* might have a high positive score (say 0.8 on a scale, if 1.0 is most positive). FiQA is challenging because the model must infer a *score* and sometimes identify the sentiment toward a specific target mentioned in the text. Benchmark results: FinBERT (Araci 2019) achieved **Mean Squared Error (MSE) of ~0.07** and a Pearson R of **0.55** on FiQA (10-fold CV) [47] [70], outperforming the previous best methods from the FiQA 2018 challenge (which had $R\approx0.40$) [47]. In other words, FinBERT improved the correlation with human scores by ~15 points (from 0.40 to 0.55). This was the state-of-the-art as of 2019. Later models likely improved this further – e.g. a 2020 study by Huang et al. reported an R of ~0.6 using a FinBERT variant – but exact numbers vary with train/test splits. FiQA being a regression task is often evaluated by MSE or R^2. Qualitatively, transformers fine-tuned on FiQA can predict sentiment scores with reasonable ranking (e.g. correctly ordering sentences from most negative to most positive), but small errors in intensity keep MSE nonzero. The dataset's size is also small, so researchers sometimes merge it with PhraseBank for extra training data (though the labels are not directly compatible, one being discrete classes and the other continuous).

- **Financial News Headline Sentiment (SemEval 2017 Task 5):** This dataset was part of a SemEval competition on fine-grained sentiment. It consists of **financial news headlines** labeled with sentiment scores in the range –1 (very negative) to +1 (very positive) towards a mentioned company [71]. For example, a headline "ACME Inc. shares plummet as CEO resigns" might have a sentiment score of -0.7. The training set had on the order of 1,000 headlines, with a few hundred for test [72]. Many participants treated it as a regression or 3-class classification by thresholding the scores. Results from SemEval 2017 showed that the best models (using ensembles of LSTMs or boosted trees with lexicon features) achieved about **0.70–0.75 correlation** with gold scores on headlines [73]. By 2020, BERT-based models surpassed these: e.g. one approach reported $R=0.81$ on the headline dataset by fine-tuning FinBERT [18]. Since this headline corpus is similar in nature to PhraseBank (short news statements), FinBERT-like models excel at it. In fact, the **"EnhancedFinSentiBERT" model achieved top performance on both PhraseBank and a headline sentiment set**, suggesting a single domain-tuned model can generalize to any financial news sentiment task [55].

- **Financial Microblogs (StockTwits/Reddit):** Another category is social media sentiment datasets. For instance, **StockTwits** (a Twitter-like platform for investors) had a labeled sentiment dataset (tweets labeled bullish or bearish). And more recently, Rahman et al. (2024) introduced **WSB-Sentiment (WSBS)**, a dataset of Reddit's WallStreetBets posts labeled for sentiment [74] [75]. These are often binary or trinary sentiment labels rather than fine-grained. FinBERT and similar models have been tested on such data, sometimes requiring domain adaptation because the language is very informal and slangy. The results are variable: FinBERT can achieve ~80% accuracy on StockTwits bullish/bearish classification after fine-tuning (per some 2021 studies), but out-of-the-box (zero-shot) it might be much lower. The WSB dataset was used to evaluate prompt-based methods (more on that in Trends section), showing that a well-prompted GPT-3.5 could reach around **70–75% accuracy** zero-shot, while fine-tuned FinBERT could exceed **80%** on the same task [74] [75].

- **Other Combined Datasets:** Some open-source projects combine multiple financial sentiment sources to create larger corpora. For example, on HuggingFace there's a dataset that merges Financial PhraseBank with a **Kaggle financial news** dataset (which contains headlines from CNBC, Reuters, etc., labeled by either an algorithm or market reaction) [76] . This merged set (~5k sentences) was used to fine-tune models like DeBERTa and yielded ~90% accuracy on a held-out test [77] . While not a standard benchmark, it shows that models trained on PhraseBank can perform well on contemporaneous news headlines (the Kaggle portion included 2020–2021 COVID-era financial news to test generalization [78]). Generally, when evaluating across datasets, **cross-domain performance drops** – a model trained on PhraseBank alone may find the FiQA or microblog data distribution different (e.g. social media sentiment is more extreme and uses slang). Thus, some research focuses on **domain adaptation techniques** to maintain robustness across such datasets (e.g. via continued pretraining on target-domain unannotated data, or multi-task learning).

In conclusion, **benchmark evaluations from 2019–2024 consistently show FinBERT and similar transformer models outperforming previous winners on all financial sentiment datasets**: PhraseBank, FiQA, SemEval headlines, StockTwits, etc. By fine-tuning or prompting, these models achieve high accuracy/correlation in each domain. For reference, FinBERT's results were: **PhraseBank ~86% acc** (macro-F1 0.84), **FiQA R~0.55** (best prior ~0.40), **Headlines R~0.8+** (best prior ~0.7), and **StockTwits ~80% acc** (where prior lexicon baselines were ~70%). The gap between domain-specific models and generic ones is most pronounced in the financial context, underscoring the need for specialization.

Emerging Trends and Future Directions

Financial sentiment analysis has matured significantly in the last five years. Current research is exploring **new frontiers beyond just improving classification accuracy** on static datasets. Key emerging trends include:

- **Multimodal and Hybrid Models:** Rather than analyzing text in isolation, there is a push toward combining **multiple data modalities**. One direction is integrating **text with numerical financial data** (e.g. stock prices, technical indicators) to build models that can not only label sentiment but also directly predict market impact. For instance, Xiang et al. (2021) proposed a deep fusion model that ingests news headlines (text sentiment via an LSTM or BERT) alongside time-series price data, improving stock index prediction [79] . Another multimodal angle is merging **news text with metadata or graphs** – e.g. using knowledge graphs of companies, or considering the network of news articles. Although images are less relevant for financial news (since articles rarely contain informative images), some work has looked at **social media sentiment with accompanying charts or memes**, which introduces image+text modeling. Overall, the trend is toward **holistic models** that understand textual sentiment in context of broader information, moving closer to how a human analyst uses both news and numbers.

- **Domain Adaptation and Continual Learning:** Financial language is constantly evolving (new terms, events, slang). Models need to adapt to shifts – for example, *"meme stocks"* or *"COVID-19 pandemic"* related sentiment was not in training data from 2014. Recent research emphasizes **continual learning** for financial models: for instance, training a sentiment model on PhraseBank, then updating it with a small set of new labels from 2020 news to adapt to pandemic-related content. One study explicitly asked *"Is domain adaptation worth it?"* for BERT in finance, and found that further pretraining on domain data yields **small but meaningful gains** on sentiment tasks, especially when the target text domain (e.g. social media vs news) differs from the source [80]. Techniques like *adaptive pretraining* (e.g. continue MLM training on recent financial news articles) and *domain adversarial training* (to make representations invariant across news/blog domains) are being tried. Another angle is **multi-lingual domain adaptation** – building models that handle financial news in multiple languages. While our focus is English, global markets require sentiment analysis in Chinese, Spanish, etc., and thus multilingual or cross-lingual transfer is an emerging challenge (some recent datasets like FinReviews cover Chinese financial news sentiment, spurring interest in cross-lingual FinBERT variants).

- **Zero-Shot and Prompt-Based Methods:** With the advent of powerful large language models (LLMs) like GPT-3/4 (2020+), there is interest in using them for financial sentiment **without fine-tuning** – i.e., via *zero-shot or few-shot prompting*. Out-of-the-box, a model like GPT-3.5 *can* perform sentiment classification if prompted, but financial nuance can trip it up. Researchers have discovered that giving the LLM **detailed task instructions** greatly improves its accuracy [74] [81]. For example, Rahman et al. (2024) introduced *Annotators' Instruction Assisted Prompting (AIAP)*, which literally feeds the same instructions that human annotators received into the GPT model's prompt [74] [81]. By explaining the task (e.g. *"You are an investor. Determine if this news is good, bad, or neutral for the stock, as our human annotators do."*), GPT-3.5's classification accuracy jumped significantly – up to a **9% improvement** in F1-score over naive prompts [75]. This kind of *prompt engineering* closes the gap between LLMs and fine-tuned models. Few-shot prompting (providing a handful of example news and their sentiments) is also used; while effective, recent studies find that **explicit instructions tailored to financial sentiment yield better consistency** [74] [82]. The implication is that future sentiment analysis might rely less on task-specific fine-tuning and more on carefully **designing prompts or using prompt-tuning** (learning soft prompts) for giant pre-trained models.

- **Prompt Tuning and Financial LLMs:** Prompt tuning refers to optimizing a small set of virtual tokens or instructions that guide a frozen language model. Given the expense of training full models, this technique is attractive in finance. We see early work on **in-context learning for sentiment**: e.g., creating a prompt template like *"Financial sentiment analysis:\nInstruction: Evaluate sentiment of the following news for stock impact.\nInput: [HEADLINE]\nAnswer:"* and possibly learning some continuous prompt embeddings to prepend. This approach can adapt a model like GPT-3 to the sentiment task with just a few parameters. Additionally, organizations are building **finance-specific LLMs** (50B+ parameters). **BloombergGPT (2023)** is a landmark example – a 50-billion-parameter model trained on a massive financial text corpus (news, filings, etc.) [83]. BloombergGPT is capable of sentiment classification among other tasks, and because it's trained on financial data, it can often perform well zero-shot or with minimal prompting on tasks like PhraseBank classification. Open-source efforts like **FinGPT** (by academic labs) also aim to adapt general LLMs (like LLaMA-2) to finance via fine-tuning on instruction datasets. For example, **FLARE** is a recent benchmark that includes financial sentiment tasks; researchers have assembled instruction-tuning data for it (with thousands of prompt-response pairs for sentiment, NER, etc.) [84] [85]. Models fine-tuned on these instructions (sometimes called *InstructFinGPT*) can achieve strong performance on sentiment without task-specific finetuning. The overall trend is moving toward **foundation models in finance** that can be *prompted* to do sentiment analysis (among other tasks) on demand, rather than training a separate classifier for each dataset.

- **Integration of Sentiment into Larger Systems:** Rather than an isolated task, sentiment analysis is increasingly used as a *component* in downstream financial systems. One trend is using model-predicted sentiment scores as features in trading algorithms or risk models. For instance, a sentiment-index is constructed from model outputs to predict stock returns; improvements in sentiment modeling directly translate to better financial outcomes in such systems [86]. Another direction is **combining sentiment with question answering** – e.g. models that can both extract a specific fact from a financial report and comment on its sentiment implication. This is part of a broader movement towards **multi-task learning** in financial NLP, where a single model might handle sentiment, named entity recognition, and even text generation (summaries or reports).

In summary, the state-of-the-art for English financial sentiment analysis is no longer just about achieving a higher F1 on PhraseBank – that challenge has largely been solved by FinBERT and successors. The focus has shifted to **robustness, adaptability, and integration**: making sentiment models that remain accurate in the wild (across different data distributions and time periods), leveraging large pre-trained models through prompts or lightweight tuning, and embedding sentiment capabilities into comprehensive financial analytics tools. These trends indicate a maturation of the field, where **financial sentiment analysis is becoming a commodity tool** that can be called upon in various ways (fine-tuned model, prompted LLM, etc.), while research pushes into new territory like multimodal understanding and continual learning to keep models aligned with the ever-changing financial landscape.

Sources:

- Malo et al. (2014) – Financial PhraseBank introduction and dataset details [3] [7]
- Araci (2019) – FinBERT thesis: model performance on PhraseBank and FiQA [35] [47]
- Rahman et al. (2024) – Prompting LLMs with annotator instructions, improving zero-shot performance [74] [82]
- Rachid (2022) – FinancialBERT model card and fine-tuning results [53]
- Lopez Roldan (2024) – Use of GPT-4 for data augmentation in financial sentiment fine-tuning [33] [30]
- Nickmuchi (2023) – HuggingFace dataset combining PhraseBank and new headlines [76]
- FiQA (2018) – Financial sentiment in the wild dataset (Task 1) [87] [47]
- SemEval-2017 Task 5 – Financial news headline sentiment task and results [73]
- BloombergGPT (2023) – Finance-specific LLM (50B) for multiple tasks [83]
- FinBERT Medium blog (2020) – Summary of FinBERT approach and motivation [67] [88]
- HuggingFace (Takala) – Financial PhraseBank data card (annotator agreement splits, etc.) [11] [3]

---

[1] [3] [4] [5] [6] [7] [8] [9] [10] [11] [12] [13] takala/financial_phrasebank · Datasets at Hugging Face
https://huggingface.co/datasets/takala/financial_phrasebank

[2] [49] [55] [62] [63] [66] GitHub - vrunm/Text-Classification-Financial-Phrase-Bank: Built a sentiment analysis model to predict the sentiment of a Financial News article. A comparative study of different optimizers used for training was done.
https://github.com/vrunm/Text-Classification-Financial-Phrase-Bank

[14] [50] [76] [77] [78] nickmuchi/financial-classification · Datasets at Hugging Face
https://huggingface.co/datasets/nickmuchi/financial-classification

[15] [25] [67] [88] FinBERT: Financial Sentiment Analysis with BERT | by Zulkuf Genc | Prosus AI Tech Blog | Medium
https://medium.com/prosus-ai-tech-blog/finbert-financial-sentiment-analysis-with-bert-b277a3607101

[16] [19] [22] [23] [26] [27] [34] [35] [38] [39] [41] [42] [43] [44] [45] [46] [47] [59] [60] [64] [65] [68] [69] [70] [80] [87]
[1908.10063] FinBERT: Financial Sentiment Analysis with Pre-trained Language Models
https://ar5iv.labs.arxiv.org/html/1908.10063

[17] [51] [52] [53] [54] [58] [61] ahmedrachid/FinancialBERT-Sentiment-Analysis · Hugging Face
https://huggingface.co/ahmedrachid/FinancialBERT-Sentiment-Analysis

[18] Financial sentiment analysis for pre-trained language models ...
https://www.sciencedirect.com/science/article/pii/S294971912500024X

[20] [21] [24] [28] [30] [31] [33] [36] [37] [40] [56] [57] [83] Financial Sentiment Analysis: Leveraging Actual and Synthetic Data for Supervised Fine-tuning
https://arxiv.org/html/2412.09859v1

[29] [PDF] Toward Text Data Augmentation for Sentiment Analysis - ArTS
https://arts.units.it/bitstream/11368/3055528/3/Toward_Text_Data_Augmentation_for_Sentiment_Analysis-Post_print.pdf

[32] [84] [85] [PDF] Large Language Model Adaptation for Financial Sentiment Analysis
https://aclanthology.org/2023.finnlp-2.1.pdf

[48] FinBERT: financial sentiment analysis with BERT - Prosus
https://www.prosus.com/news-insights/group-updates/2020/finbert-financial-sentiment-analysis-with-bert

[71] SemEval-2017 Task 5: Fine-Grained Sentiment Analysis on ...
https://aclanthology.org/S17-2089/

[72] NLG301 at SemEval-2017 Task 5: Fine-Grained Sentiment Analysis ...
https://openreview.net/forum?id=hrpwpZVTkT

[73] UW-FinSent at SemEval-2017 Task 5: Sentiment Analysis on ...
https://paperswithcode.com/paper/uw-finsent-at-semeval-2017-task-5-sentiment

[74] [75] [81] [82] [86] Evaluating Financial Sentiment Analysis with Annotators' Instruction Assisted Prompting: Enhancing Contextual Interpretation and Stock Prediction Accuracy
https://arxiv.org/html/2505.07871v1

[79] A deep fusion model for stock market prediction with news headlines ...
https://link.springer.com/article/10.1007/s00521-024-10303-1