FinLex: An Effective Use of Word Embeddings for Financial Lexicon Generation

Sanjiv R. Das[1,2,*], **Michele Donini**[1], **Bilal Zafar**[1], **John He**[1], **Krishnaram Kenthapadi**[1]
[1] AWS (Amazon Web Services)
[2] Santa Clara University

Abstract

We present a simple and effective methodology for the generation of lexicons (word lists) that may be used in natural language scoring applications. In particular, in the finance industry, word lists have become ubiquitous for sentiment scoring. These have been derived from dictionaries such as the Harvard Inquirer and require manual curation. Here, we present an automated approach to the curation of lexicons, which makes automatic preparation of any initial word list immediate, which can then be further curated. We show that our automated word lists deliver comparable performance to traditional lexicons on machine learning classification tasks. This new approach will enable finance academics and practitioners to create and deploy new word lists in addition to the few traditional ones in a facile manner.

1 Introduction

Text-based numerical scores and features such as sentiment, readability, positivity, negativity, riskiness, and litigiousness play an important role in many financial research and practical applications (see Loughran and McDonald (2020) for a comprehensive review). The existing practice is to derive word-based features from the input text by applying finance-specific dictionaries such as the Loughran-McDonald (LM) word lists.[1] Word scoring based numerical features are widely used in regression analysis of regulatory filings, news articles, tweets, etc. It is a testament to the success of this approach that a vast literature in accounting, economics, and finance has exploited these variables, amounting to hundreds of papers in the last two decades. Gentzkow et al. (2019) provide a survey of applications and the motivation for the use of text and scoring methods. Other comprehensive surveys of textual analysis in finance include: Li (2010); Das (2014); Kearney and Liu (2014); Loughran and Mcdonald (2016). These papers all make the point that lexicons are effective in representing various financial concepts and the numerical scores of documents are related to financial outcomes and corporate performance, justifying the original ideas in (Loughran and Mcdonald, 2011). We complement the hand-curated lexicons with an automated approach.

Several applications benefit from the use of word lists in natural language processing (NLP). NLP helps in credit analysis of firms using SEC filings and news (Bodnaruk et al., 2015; Bonsall et al., 2017; Ertugrul et al., 2017). Routledge (2019) presents an example of NLP in asset management. Corporate performance is related to textual features such as sentiment, readability, tone, size, risk, and uncertainty (see Antweiler and Frank (2004); Das and Chen (2007); Hoberg and Phillips (2016); Bach et al. (2019) and the references there-in). Differences in the text of filings across quarters and years contains predictive value as shown in Cohen et al. (2020). Several academic studies and industry whitepapers have shown the value of sentiment scoring and text variables (e.g., Hafez et al. (2020b,a); Cong et al. (2019b,a); Chebonenko et al. (2018)). Recently Twitter feeds have become important sources of text that may be related to financial outcomes, as discussed in (Blankespoor et al., 2014; Elliott et al., 2018). Bartov et al. (2017) show that the aggregate opinion from individual tweets predicts firms' forthcoming quarterly earnings and announcement returns. Chen et al. (2019) construct a crypto-specific lexicon and find a relation to cryptocurrency returns. Thus, it is clear that bags of words may be constructed for special purpose financial concepts, various markets, and media sources (chat rooms, discussion boards, SEC filings, tweets,

Contacts: Sanjiv Das (srdas@scu.edu, Michele Donini (donini@amazon.com), Bilal Zafar (zafamuh@amazon.com), John He (hezhijia@amazon.com), Krishnaram Kenthapadi (kenthk@amazon.com).

[1] https://sraf.nd.edu/textual-analysis/resources/

news, etc.).

The key is where the bag of words comes from. The approach in the extant literature has been to rely on experts to create these bags. But this introduces a degree of subjectivity and a lack of robustness. Moreover, as the number of concepts increase, and are represented differently over time – perhaps due to evolution of language, economics (e.g., rules may change what is subject to litigation), and as expertise on different topics or even the same ones spread, the need arises for discipline and replicability in how the bags of words are constructed. This paper's purpose can be viewed as bringing discipline to the process of creating bags of words for different concepts. First, the approach replaces subjective choices of humans with a replicable algorithm. Since the algorithm is completely described and is easy to implement, it is replicable. Second, it can be refined by future modelers and is therefore more easily developed and adapted. Third, it does not do away with the domain experts– it simply gives them a superior baseline on which their expertise can be used to refine the results. Finally, the role of experts is also more objective as what experts impose their views on becomes transparent. This makes the entire bag of word list generation transparent in ways that it has not been before.

Motivated by the need for automatic creation of word lists suitable for different financial applications, we present FinLex, a simple and scalable approach to generate financial word lists using pretrained word embeddings, requiring no manual curation whatsoever. Our approach facilitates generating word lists that provide support both *for* as well as *against* a concept. A similar approach was used recently to create a corporate culture lexicon in (Li et al., 2020). In our approach, the user simply provides a pair of words that are either synonyms or antonyms. If the words are synonyms, we generate two word lists with embeddings that are closest to the two words, intersect these lists with a dictionary to keep only the ones that are valid words, and then return the union set of both word lists. Using the intersecting dictionary is an additional approach to triage the word lists before manual curation as it catches spelling errors, non-English words, etc. If the words are antonyms, we generate two word lists with embeddings that are closest to the two words, intersect these lists with a dictionary to keep only the ones that are valid words, and then return two separate word lists. If a word appears in both lists, then we keep the word only in the list in which it has highest similarity. In short, with synonyms, the algorithm returns a single list (support *for* the concept) and with antonyms, it generates two lists (support *for*, as well as against *against*, the concept).

We demonstrate that these automated word lists (denoted as financial lexicons) are remarkably good at generating words related to the concepts in the synonym or antonym pairs. We note that there are only seven Loughran-McDonald (LM) word lists, for the following word concepts: negative, positive, uncertain, litigious, strong modal, weak modal, and constraining. In contrast, our approach enables the creation of lexicons for financial concepts beyond those currently supported by the LM word lists, and thus addresses a practical need for the financial services industry. We illustrate this benefit by developing word lists for additional concepts such as fraud, negligence, riskiness, and safety. We also ran classification tasks using our automated lexicons and compared the results for the same task undertaken with the current state of the art LM word lists. We find comparable classification results, thereby supporting the value of the LM word lists as well as that of the algorithmically generated financial lexicons.

The rest of this paper proceeds as follows. Section 2 presents the algorithm and its output. Section 3 compares the automated financial lexicons with the hand-curated LM word lists on classification tasks. Concluding discussion is presented in Section 4.

2 Methodology

2.1 Historical perspective

The Loughran-McDonald (LM) word lists were derived from the Harvard Inquirer (Stone et al., 1966; Kelly and Stone, 1975) word lists.[2] These word lists (lexicons) are described in Loughran and Mcdonald (2011); Bodnaruk et al. (2015); Loughran and Mcdonald (2016); Loughran and McDonald (2020). These authors painstakingly adapted these word lists to finance applications by specifically removing words that were ordinarily positive or negative in the English language but not so in finance. For example, the word "liability" has a negative connotation in normal language, but is neutral in finance, as a liability is an accounting

[2] http://www.wjh.harvard.edu/~inquirer/

term. Loughran and Mcdonald (2011) point out that almost 3/4 of the words in the Inquirer negative word list were not deemed as such in finance. Indeed, while financial returns have no relation to the Harvard IV-TagNeg (negative words) list, they do have a relation to the recast LM negative word list. LM further developed seven canonical lists for text "tone"—the 2018 file has 2355 negative, 354 positive, 297 uncertainty, 904 litigious, 19 strong modal, and 27 weak modal words. Recently, another category was added, for 184 constraining words.[3] The quality of these word lists is high and extensive validation of these lists has been undertaken by establishing correlations to economic phenomena in the literature, as cited above. Correlations with text scores from SEC filings are related to financial returns, seasoned equity offerings, news articles, etc.

The approach in this paper offers an automated alternative to curated word lists, enabling the creation of word lists for machine learning and econometric analysis with no manual intervention whatsoever. This enables generation of word lists for categories beyond the ones currently used in practice. For example, we may wish to score documents for words related to the concepts of "fraud" and "negligence". Another example is to create a list of "safe" words as a counterpoint to "uncertainty/risk" words.

Because we stipulate no manual intervention, these word lists will surely include errors. However, we note two important considerations in this regard: (i) these word lists can be further curated manually to remove any irrelevant words. Such filtering steps are likely to involve much smaller manual effort than constructing the lists from scratch; (ii) if the word lists provide strong predictive performance, they will arguably serve their purpose. As shown in Section 3, these automated word lists deliver comparable performance to the LM word lists on example classification tasks, even without manual curation.

2.2 Algorithm

The approach (denoted as FinLex) is as follows. It takes as input a pair of words (w_1, w_2). These words may be synonyms or antonyms. The procedure is as follows:

1. **Embedding model selection:** Retrieve a large set of word embeddings from a pretrained word embedding model (examples will follow shortly). Word embeddings are numerical vectors of fixed dimension D (usually 300) that represent each word, thus, each word can be imagined as a point in D-dimensional space. The collection of unique words for which embeddings are generated is called the "vocabulary" and the size of the vocabulary is denoted by V. Hence, the collection of word embeddings can be represented in a matrix of size $V \times D$. Think of this as a projection of V words into D dimensions. Words related to the same concepts and context will reside closer to each other in this vector space. Standard sources are FastText from Facebook,[4] Global Vectors (GloVe) from Stanford,[5] and word2vec from Google[6]. Each of these uses slightly different approaches to generate word embeddings from large text corpora (e.g., BOW, SkipGram, GloVe, etc). We used FastText for our source of embeddings, (Mikolov et al., 2017; Grave et al., 2018), based on the original work by Mikolov et al. (2013).[7] These embeddings are usually vectors of size 300 but we have reduced them down to size 100 using FastText's built-in dimension reduction to make the computation efficient.[8] We note that this is a first approach and several variations may be tried for embedding size D and source/technique for generating embeddings. More recent Transformer models like BERT (Devlin et al., 2019) use larger embedding sizes ($D = \{768, 1024\}$), but with these models, the same word can result in different embeddings based on the surrounding context. We leave the coupling of contextual embeddings with bag of words (BoW) models to a future investigation.

2. **Selecting similar words:** For w_1 and w_2 find the set of K most similar words in the embedding space, using cosine similarity, denoted

[3] https://sraf.nd.edu/textual-analysis/resources/#Master%20Dictionary

[4] https://fasttext.cc/docs/en/crawl-vectors.html

[5] https://nlp.stanford.edu/projects/glove/

[6] https://code.google.com/archive/p/word2vec/

[7] For the code, see https://fasttext.cc/docs/en/crawl-vectors.html, see also https://mb-14.github.io/tech/2019/02/19/word-embeddings-js.html

[8] See also Joulin et al. (2016)

$w_S(w_i), i = 1, 2$. We set $K = 1000$ for the examples in this paper. This is a large number of similar words to examine and represents an upper bound on how big a lexicon can be. Moreover, after intersecting the set of most similar words with an English dictionary, many of them drop out, so we tend to get a smaller set of words, somewhat below 1000. We examined setting $K = 5,000$ and found that the additional words after the first 1000 were less conceptually similar to the input words (w_1, w_2). We note that we get 1000 words each for w_1 and w_2, so we are collecting a very large number of similar words compared to the size of curated lists in the literature. The LM word lists are of the size we obtain, though they also contain different versions of the same word, so tend to be a little longer. As discussed later, when stemming is applied, the LM lists become much shorter, so $K = 1000$ is a comfortable upper bound on the word list size.

3. **Filtering for standard words:** We then lower case the words and intersect the set with a list of standard English words, denoted as set w_E. In our case, we simply used the words from Harvard Inquirer, but any word list may be used, e.g., the MIT word list.[9] Therefore, we have the new set $S(w_i) = w_S(w_i) \cap w_E, i = 1, 2$. This ensures a clean list of words that are related to the concept words w_i.

4. **Merging:** Here we have two cases, depending on whether the pair of words are synonyms or antonyms. Using a pair of words is a new idea and offers greater inclusion of concepts and improved accuracy in lexicon generation. (1) *Synonyms*: If w_1, w_2 are synonyms, we then combine the two sets so that the output is a single set of words, $S(w_1) \cup S(w_2)$. This engages a concept more broadly if there are two words that are conceptually related, e.g., fraud and negligence. We note that if we want a word list based on only a single word as in previous applications in the literature, then we would just set $w_1 = w_2$. (2) *Antonyms*: If w_1, w_2 are antonyms, then the output is two sets of words, $S(w_i), i = 1, 2$, where each set is independently generated using cosine similarity. If a word occurs in both $S(w_1)$ and $S(w_2)$, then we retain it only in the list where it is closer to w_1 or w_2, and drop it from the other list. This ensures a word cannot be common to two opposing concepts. This is especially important when using word embeddings, because it is possible that words that are opposite in meaning to the concept word may also reside nearby in embedding space, since they are related to the concept word, even though they have an opposing connotation. For example, if the antonyms are "war" and "peace", the word "conflict" may appear in both lists and if it closer to "war" (as it hopefully should be), then it will be removed from the list based on the word "peace". This is a simple check and balance on automated lexicon generation as it helps triage misclassified words.

This procedure creates a single word list if w_1, w_2 were synonyms and two word lists if they were antonyms. We do not remove stopwords, nor do we stem the words, so that multiple forms of the same word may appear, as is the case with the LM word lists. Note that this procedure can be easily extended to cases where multiple words are used to represent a concept, i.e., multiple words describing each concept in the antonym pair.

[9] https://www.mit.edu/~ecprice/wordlist.10000

Table 1: Examples of word lists generated from `FinLex`. For each pair of words, we denote them as antonyms or synonyms, and also present the number of items in the derived word list.

Pair Type	w_1, w_2	No. of words
Antonyms	positive, negative	286, 259
Antonyms	risk, safe	240, 215
Synonyms	litigious, litigation	198
Antonyms	uncertainty, certainty	175, 257
Synonyms	fraud, negligence	269
Antonyms	fair, unfair	285, 230

2.3 Examples

We created several word lists using this procedure. Table 1 displays the number of words in each list based on the pair of concept words.

In Figure 1 we display the negative and positive word lists generated by our procedure. As we see, the words are quite well-related to the concept word. There are clearly words that are out of place, such as in the negative word list: neutral, causal, truth, imply, certain, normal. In the positive words

list there are incorrect words such as passive, stress, selfish, and hesitant. These words may naturally be hand-curated out with little manual effort. However, in this particular study, in order to show the potential of keeping the process fully automated, we do not include manual interventions.

As an example of a synonym pair of words, we present results for the "fraud, negligence" pair in Figure 2. Once again, the set of words is remarkably well-related to the concept words.

In order to examine the effect of Step 3 in the algorithm, Table 2 displays the words obtained from the root word "positive" that are excluded by intersecting with a dictionary. The primary effect of this step is to remove words that have incorrect spelling. As an example, see the top words (based on word vector cosine similarity to the root word) that are excluded. There are a large number of misspelled words that are in this list (e.g., postive, postiive, etc.), but there are some excluded words that may have been retained (e.g., stronger, synergy, etc.). Misspelled words occur because the corpus of word embeddings is based on Wiki text which also contains several words that are incorrectly spelled, yet occur in the context of the concept word. While the misspelled words will be caught by intersection with any dictionary, the words that may have been retained will occur if a bigger dictionary is used. The dictionary we used had only 8,642 words, whereas WordNet[10] has a list of words that numbers 147,306. If longer word lists are used, the excluded words would be less likely to include correct words, while being just as effective in removing misspelled words.

We also examined the overlap between the LM word lists and those from FinLex. These are shown in Table 3. It is interesting that the overlap in the word lists is not large, though it tends to increase a little after stemming.

It is useful to examine the overlap further. As an example, we look at the "uncertain" words and note that after stemming, the Loughran-McDonald (LM) lexicon has 136 words (down from 297 pre-stemming). The FinLex positive word list has 170 words (down from 174 pre-stemming). Post-stemming, the intersection set of the LM and FinLex word lists is 15 words. These words are shown in Table 4. We chose the uncertain word list as it had the smallest overlap and so may reveal interesting differences between human and machine curation of lexicons.

We can see the small set of intersection words all connote uncertainty. What is interesting is the large set of words that are in the LM lexicon and not in FinLex, and vice versa. Both these lists quite clearly have words that we would agree do relate to the concept of uncertainty. So why such a small overlap? It is possible that the mechanisms are different. Human curation may be based on mental retrieval or through the use of a thesaurus, an approach that is direct. The machine curation approach exploits textual context from word embeddings, so is an indirect approach in that words that do not appear together but appear in similar contexts tend to have similar word embeddings. This suggests that a combination of human and machine lexicon curation may be useful. In any case, further curation of the machine generated lexicon by humans is probably useful to remove words that do not belong in the word list.

3 Performance of word lists on classification tasks

3.1 Datasets

We considered various datasets for an assessment of word lists as features for classification analysis. For each document in the datasets, we compute the fraction of words in the document that are in a given word list, and add this as a column (numerical feature) to the dataset. For example, for negative words from LM, we create a separate column. Likewise for every word list from LM and FinLex.

The first dataset used to assess the performance of features based on word lists for a classification task is the Financial Phrase Bank (FPB).[11] The dataset comprises financial news headlines with three label (the discrete y-variable) categories for sentiment: negative, neutral, and positive. The sentiment scores are based on manual labeling by several annotators. The dataset is an amalgam of sub datasets. One sub dataset contains 2264 news headlines on which all annotators agree (negative 13%, neutral 61%, positive 25%). This was added to more sub datasets with additional news headlines with annotator agreement ranging over 75%, 66%, and 50%. In total, across all sub datasets there are 4846 news headlines with sentiment levels: negative 13%, neutral 59%, positive 28%). Therefore,

[10] https://wordnet.princeton.edu

[11] https://www.kaggle.com/ankurzing/sentiment-analysis-for-financial-news

Negative Words

negative detrimental worsen hurtful slight mild anomaly thoughtless guilt effect cynicism threshold disruptive fear dislike incorrect fixate assumption risky rejection severity restriction discriminatory strain intolerable neutral aggravate insignificant unfortunate illogical bad confusion ambiguous burden contradictory wishful corrosive separation distort tolerate superficial factor point injurious shock relapse predictable misuse abnormal risk appreciable pernicious irrational consequence disputable unfair unfavorable inconsistent apathy disproportionate concern alarming pessimistic hazard lack decline discernible trivial disagreeable uninformed deplorable undefined denial horrible tolerable allowable apparent lag ineffective zero excessive ignore indifferent wrong undue likely breakdown incompatibility repulsive causal partial erroneous contradiction truth unkind discrepancy uneasiness diminish touchy worst marginal tolerant stigma correlation bothersome antagonism uncertainty weakness uncomfortable delusion antagonistic differential unnecessary immoderate imprecision untrue implication upset questionable partisanship pessimism false uneasy disinterest depression drastic presumption unreasonable equivocal common anomalous imply deficient lessen certain unavoidable deviation normal indicative disregard nonspecific stupidity indifference extreme discomfort unimportant precipitate aversion behavior likelihood perception reaction substance indirect inherent unfavourable negligible predominant unjustified loss habitual irritation persistent distortion avoidance inhibition insufficient severe disadvantageous inconsistency indication regrettable positiveness cause obvious depress not worse terrible retardation aggravation racial positive unhealthy same exaggerate inconvenient misunderstanding ignorance problem positivity leakage discrimination undesirable weak cumulative ambivalent prejudicial extent sensitive negate contrary blame disbelief taint worry discordant dependence unrealistic hostility suffer interference affect ambiguity prejudice untruth discrepant regardless nor m justifiable tolerance inexplicable intolerance fearful odds inevitable nothing vagueness latent adverse disagreement inadequate persist compromise prevalent destructive harmful harsh symptom harm improper tendency unpleasant opposite troublesome acceptable insidious headache disorder regress similar distrust decrease fault unnatural benign nasty defect exclusion

Positive Words

positive adequate passive seek presence supportive cohesive creative undecided importance candid stress forceful individual natural trust alone only confidence primary given significant selfish constructive affirmation radical hesitant pure emotion anxiety better compassionate gradual unique joyful moderate instinctive rapport welcome upbeat doubt value feel personal subjective truthful meaningful optimism momentary motivation clarity reflection individuality therapeutic barrier expectation overall seriousness increase recognition permanent tremendous mental social perspective emotional proportionate compassion confident choice diversity functional affinity necessarily great strong expect imperative hesitancy unquestionable stability unexpected outlook conscious important faith potency distraction genuine safe treatment relief reduction direct optimal reassurance significance sober kind outcome harmonious selective appropriate tension cautious true influence reasonable efficacy altruistic measurable deal right vital mood empathy cohesion improvement willingness awareness less priority overcome alternative ethical undeniable contagious reliance potential conclusive proactive strength continuous certainty preference apprehensive remarkable solid urgency healthy well-being motivated impulsive hopeful serious thoughtful sensible fundamental favourable optimistic effectiveness any noteworthy active constant immediate difference attention goodwill particular energetic self subtle negative best that profound initial psychological sympathy beneficial universal exceptional free impression benefit definitely response possible consistent lead possibility contribute fruitful reflect reciprocal apathetic sane minimal greatest aspiration attitude mutual more gain evident happy interpersonal dependent definite result satisfactory noticeable pressure equal equitable closeness unity depend upfront mindful relevance bond perfect tangible specific tentative real focus dramatic short-term cohesiveness affirmative sympathetic good unwavering crucial change aggressive intensity patient always unequivocal growth apprehension yet spontaneous hindrance dynamic deliberate unconditional objective effective belief sense ecstatic sufficient surprise determination personality attainable understandable success amicable opportunity rational sincere sensitivity favorable purposeful critical commitment clear share commendable calmness reactive overwhelming anxious decisive intangible absolute concentration long-term achieve receptive balance physical anxiousness essential adversity respect insight acceptance impact substantial

Figure 1: Negative and positive word lists generated by the procedure. This is an example of antonyms generating two lists.

Fraud, Negligence Words

fraud abandonment abatement abuse accident accidental accusation adjudication adulteration alienation allegation answerable anti-trust antitrust apathy applicant appraisal apprehension arrogance assessor assurance attorney auditor avarice avoidance bankruptcy beneficiary blackmail blindness blunder bogus breach broker burglary calamity callous careless carelessness cartel circumstance cleanliness coercion collusion commissioner compensation competence complaint compliance complicity concern condemnation condition confiscation conspiracy contempt conviction corruption counterfeit credibility credit crime criminal cruelty culpable danger debt debtor deceit deceitful deception deceptive defect defendant deficiency denial deplorable despicable deterrent diligence disability disclosure discrimination disgrace dishonest disobedience dispute disregard disruption dissatisfaction distress employer employment entanglement ethics evasion exclusion expediency failure fake false falsehood fault fiasco financial fraud fraudulent greed grievance guarantor guardianship guilty harassment hardship harm hazard helplessness ignorance illegal illegality illness immaturity immigration immorality impartiality impediment impostor imposture improper inadequacy inadequate incapacity incompetence incompetent inconvenience indemnity indictment indifference ineffectiveness inefficiency inequity inexcusable infirmity infringement ingratitude inhumane injurious injury injustice inspection insubordination insurance integrity investigation involvement irresponsible jeopardy judgment landlord law lawyer legal legitimate liability liable liar liquidation litigant litigation maladjustment malice malicious management manipulation manslaughter misfortune mishap mismanagement mistreatment misunderstanding misuse monetary money monopoly mortgage motive murder neglect negligence negligent nuisance obstruction occupational offence offender omission oversight owner payer payroll pecuniary penalty petitioner plaintiff plea policy predicament prejudice prejudicial presumption profit profiteer prosecution prudence publicity reckless recklessness recompense recourse redress refusal rejection remorse reparation reprehensible reputation responsibility responsible restitution risk robbery roofer safety satisfaction scandal scrutiny selfishness slander solicitor soundness sting stupidity subsidy suspicion tax taxation taxpayer tenant terrorism theft threat trade transaction treachery trespass unfair unhappiness unjust unjustified unlawful unsatisfactory unscrupulous unsoundness untimely vexatious victim violation welfare willful wrongful

Figure 2: Fraud and negligence combined word list generated by the procedure. This is an example of synonyms generating two lists.

Table 2: Word Comparison for Step 3. The table shows the top 50 included words and the top 50 excluded words after intersecting with a dictionary to remove erroneous words.

Top 50 excluded words	postive postiive postitive positve positivie positivity postivie negativity reaction negatively stronger measureable synergistic lasting reactions greater quantifiable possitive tolerance negative- affective ositive negative.it feeling catalyst euphoric intentional surprising empathic unalloyed perception interaction self-statements unfavorable contributive drastic synergy posotive happier positvely concern paradoxical negtive non-anxious detrimental self-limiting self-compassionate pessimistic unmeasurable strongest
Top 50 included words	positive surprise amicable reflection concentration passive unwavering cohesion cohesive calmness radical hopeful affirmation unique individual real creative welcome decisive solid reactive bond affinity optimism urgency unity efficacy vital confidence contagious mindful deliberate physical sincere significance exceptional definite self forceful conscious hesitancy essential permanent noteworthy importance compassion focus proactive negative any

Table 3: We examine the overlap in words for both LM lists and from FinLex. We report the number of unique words in each list before and after stemming, and the number of overlap words. In each column, we show two numbers, the first is before stemming and the second one is after stemming.

Word Type	LM	FinLex Size	Intersection Size
positive	354, 151	285, 266	29, 35
negative	2355, 915	258, 237	97, 100
uncertain	297, 136	174, 170	14, 15
litigious	904, 456	197, 194	17, 17

the labels are imbalanced across the three categories. The dataset was prepared by Malo et al. (2014) and used in Araci (2019).

The second dataset used is the Disaster Tweets dataset (https://www.kaggle.com/vstepanenko/disaster-tweets). This dataset comprises 11,370 tweets, of which 19% relate to disasters and the others do not. Therefore, the labels are also imbalanced for this dataset. This dataset supports a binary classification exercise, where we fit machine learning models to predict whether a tweet relates to a disaster or not.

The third dataset is the Reddit News dataset (https://www.kaggle.com/aaron7sun/stocknews) from the Reddit World News channel. This contains the top 25 (based on Reddit rankings) news headlines for each day with a binary label as to whether the Dow Jones Industrial Average Adjusted Closing value rose or stayed as the same. The dataset comprises 1989 days. For each day we merged all 25 news headlines to make a single document to accompany the label. This dataset is then used to fit machine learning models for binary classification.

3.2 Classification experiments

We fitted classification models on 4 different feature representations of the text. This creates a horse race between different lexicons by fitting models to 4 different datasets.

- First, we used a Term Frequency - Inverse Document Frequency (TFIDF) representation of each text document for classification. This standard approach uses all the words in the articles (not just numerical scores of word list counts) and is a useful baseline. If we have N text articles in a dataset and a vocabulary of V unique words across all the articles, then the frequency count of words (terms) in each article may be tabulated in a term-document matrix of dimension $V \times N$. For each row of this dataset, i.e., term-frequency (TF) counts of the word in the row for all documents in the columns, we divide the value in the matrix by the number of documents that contain the word, i.e., multiply by inverse document frequency (IDF). This results in the TF of words that are less common across documents becoming overweighted in each column relative to words that occur more often across documents. We call the adjusted matrix the TFIDF matrix. The transpose of this matrix has documents on the rows and words on the columns.

Table 4: Examples of overlap and non-overlap words from the LM and FinLex word lists for the "uncertain" word list. These words are compared after stemming.

Common words in LM and FinLex	ambigu anomal anomali cautiou confus depend imprecis instabl predict risk turbul uncertain uncertainti unexpect unpredict volatil
Words in LM, not in FinLex	abey almost alter anticip appar appear approxim arbitrari arbitrarili assum assumpt believ cautious clarif conceiv condit confusingli conting could crossroad destabil deviat differ doubt exposur fluctuat hidden hing improb incomplet indefinit indetermin inexact intang likelihood may mayb might nearli nonassess occasion ordinarili pend perhap possibl precaut precautionari predictor preliminari preliminarili presum presumpt probabilist probabl random randomli reassess recalcul reconsid reexamin reinterpret revis riski riskier riskiest roughli rumor seem seldom seldomli sometim somewhat somewher specul sporad sudden suddenli suggest suscept tend tent uncertainli unclear unconfirm undecid undefin undesign undetect undetermin undocu unexpectedli unfamiliar unforecast unforseen unguarante unhedg unidentifi unknown unobserv unplan unprov unproven unquantifi unreconcil unseason unsettl unspecif unspecifi untest unusu unwritten vagari vagu vaguer vaguest vari variabl varianc variant variat
Words in FinLex, not in LM	abandon abrupt absenc acrimoni advers anger anxieti anxious apathi aris avoid bafflement bewilder breakdown calam catastroph certainti chang chao coincid collaps complex concern confid conflict congest consequ constraint correl crisi cynic declin deficit delay depress despair disagr disappoint discont discord discourag disillus disinterest disrupt dissatisfact distrust disun econom emot empir entangl equilibrium exasper exhaust failur fallout falter fatigu fear financi fraught frustrat gloom gradual hardship hesit horizon immin impact impass impati impedi impetu inadequaci incipi incompet inconsist indecis indic indiffer ineffect ineffici inequ inevit influenc insecur interdepend interrupt irregular loss mismanag mistrust misunderstand mitig momentum neg nervous optimist outcom overcom paralysi partisanship percept persist pessim pessimist postpon preoccup procrastin progress puzzlement quantit readjust recess reckless reflect reject relianc resent restless retrench rise sad scarciti sever shift short-term shortcom slump stabil stalem stress strife stubborn tension trend turmoil unavoid uneasi unfavor unfavour unforeseen unhappi unreli unrest unsteadi unsur upheav upturn urgenc vex vulner weak weari worsen

Each row of this (sparse) matrix represents a vector of numerical features for each document that can be used for machine learning to fit a model to predict the label. This is a large collection of variables for prediction, known as a feature set, and is much larger than a few columns of data that contain the text scores from either the LM model or the FinLex model, described next.

- Second, we used the Loughran-McDonald word lists for negative, positive, uncertain, and litigious words to count the fraction of words in each list that occur in each document. This converts each document into 4 numerical columns of data. Lexicon-based scoring is one way of converting text into numerical values that may then be used for data analysis, either using econometrics or machine learning. This representation of the text has a very small feature set of just four variables (positivity, negativity, uncertainty, and litigiousness). If this model is able to come close to the prediction performance of models that use all the text with the TFIDF representation, then it justifies the use of word lists for practical applications as has been done by academics and practitioners in the finance community for quite some time.

- Third, for comparison with the LM feature set, we used the same four features that were generated by the FinLex algorithm. We fit machine learning models to these 4 variables.

- Fourth, we use the previous FinLex dataset and enhance it with additional columns for new lexicons, based the words fraud and negligence, safe, risk, certainty, uncertainty, fair and unfair. We call this set of variables the FinLex+ model.

All 4 representations of the text are then fitted to an ensemble of machine learning models. We used an AutoML package called AutoGluon for classification (Erickson et al., 2020). AutoGluon is an open-source AutoML framework invoking a single line of Python to train machine learning models on an unprocessed tabular dataset (numeric plus text). Unlike existing AutoML frameworks that primarily focus on model/hyperparameter selection, AutoGluon-Tabular succeeds by ensembling multiple models and stacking them in multiple layers.

Table 5: Comparison of classification models using word lists. MCC stands for the Matthews Correlation Coefficient (range from -1 to $+1$ with positive values supporting predictive ability), and the precision, recall, and F1 scores are weighted averaged across the label categories. The four models are based on TFIDF word vectors, four numeric features computed from the LM word lists, the same four numeric features based on matching FinLex word lists. Numbers in parenthesis are standard deviations from 5 repetitions of the same experiment, all numbers are reported on the test dataset.

FPB All Agree dataset (2264 obs, 80:20 split)				
Metric	TFIDF	LM	FinLex	FinLex+
Accuracy	0.810 (.01)	0.650 (.03)	0.630 (.02)	0.701 (.02)
MCC	0.637 (.03)	0.279 (.04)	0.188 (.03)	0.400 (.04)
Precision	0.798 (.01)	0.627 (.04)	0.572 (.03)	0.650 (.05)
Recall	0.810 (.01)	0.650 (.03)	0.630 (.02)	0.701 (.02)
F1	0.796 (.01)	0.592 (.04)	0.534 (.03)	0.657 (.03)
FPB All dataset (4846 obs, 80:20 split)				
Metric	TFIDF	LM	FinLex	FinLex+
Accuracy	0.722 (.01)	0.623 (.01)	0.598 (.02)	0.637 (.01)
MCC	0.458 (.03)	0.243 (.05)	0.112 (.03)	0.244 (.01)
Precision	0.721 (.02)	0.599 (.02)	0.516 (.02)	0.588 (.02)
Recall	0.722 (.01)	0.623 (.01)	0.598 (.02)	0.637 (.01)
F1	0.694 (.01)	0.585 (.02)	0.488 (.01)	0.577 (.01)
Disaster Tweets dataset (11370 obs, 80:20 split)				
Metric	TFIDF	LM	FinLex	FinLex+
Accuracy	0.878 (.01)	0.820 (.01)	0.819 (.00)	0.816 (.00)
MCC	0.520 (.03)	0.032 (.04)	0.009 (.01)	0.069 (.02)
Precision	0.865 (.01)	0.736 (.04)	0.709 (.02)	0.758 (.03)
Recall	0.878 (.01)	0.829 (.01)	0.819 (.00)	0.816 (.00)
F1	0.861 (.01)	0.742 (.01)	0.749 (.00)	0.747 (.01)
DJIA Reddit News dataset (1989 obs, 80:20 split)				
Metric	TFIDF	LM	FinLex	FinLex+
Accuracy	0.512 (.02)	0.507 (.02)	0.527 (.02)	0.521 (.02)
MCC	0.004 (.03)	-0.018 (.05)	0.043 (.02)	0.031 (.04)
Precision	0.509 (.02)	0.450 (.10)	0.527 (.01)	0.521 (.02)
Recall	0.512 (.02)	0.507 (.02)	0.527 (.02)	0.521 (.02)
F1	0.447 (.05)	0.437 (.09)	0.495 (.05)	0.476 (.06)

The same models were ensembled across all experiments. These models comprised feed-forward neural net, kNN, random forest, extra trees, Light GBM, XGBoost, and CatBoost. This supports agnostocity towards hyper-parameter selection.

AutoGluon is an advanced approach to building the best machine learning models and then ensembling them together to obtain a better model. We provide some brief details here. The numerical variables that represent the text (the X independent variables) and the labels (the Y dependent variable) are used to train each of the machine learning models mentioned above. (Think of this as fitting multiple separate nonlinear regression models.) These predicted values from each model are added to the X variables and then the models are refitted to the extended set of independent variables. The predictions from each of these models are then weighted to create a composite prediction, where the weights on the models are determined by a neural net that finds the optimal model weights. This entire approach is known as "stack-ensembling" and is an accurate and effective way of pooling the wisdom of the crowd of individual ML models.

Why use the AutoGluon ensemble approach to fit machine learning models to the 4 datasets (representations of the text)? By ensembling and weighting different machine learning classifiers on the datasets, we find the best model for each dataset. It is possible that XGBoost does better for the FinLex dataset, whereas Random Forest does better for LM. If we only fitted the 4 datasets to XGBoost, we would find that FinLex outperforms LM. By ensembling via AutoGluon, we ensure that each dataset is able to find its best prediction model automatically, and this makes the comparison across different lexicons fair.

Training and performance evaluation of all 4 feature sets is reported in Table 5 in 4 columns: (i) the TFIDF features from the text of the datasets, (ii) just the columns of lexicon scores (percentage of text from the lexicons) from the LM word lists, (iii) just the columns of lexicon scores from the FinLex word lists, and (iv) just the columns of lexicon scores from the FinLex+ word lists. Five metrics are reported (see the rows of the table): (1) Accuracy, i.e., the percentage of correct predictions, (2) Matthews Correlation Coefficient (MCC), which ranges from -1 to $+1$ where scores above 0 are indicative of classification ability, (3) Precision, i.e., the number of predictions of a particular category that the model gets correct, this focuses on how few the false positives are, (4) Recall, i.e., the number of cases in each class that are correctly predicted by the model, this focuses on how few the false negatives are. (5) Finally, the F1 score, which is the harmonic mean of Precision and Recall. The results for TFIDF are the best because it is based on the full text, not just the word-based scores. The financial text analysis literature has focused on the word-based scores so they could run regressions, but with the growing use of machine learning, it is clearly better to use text representations as we see the results are better when the full text is used.

Table 5 presents the results, showing average and standard deviation of all metrics on the test samples over 5 repetitions using 80% of the data

for training and the rest for testing. We note the following results. First, the classifier based on just the numerical LM scores performs quite well relative to the TFIDF classifier, especially when the classification problem is harder (lower accuracy), which provides justification for the approach taken in the empirical finance literature and in practice. Second, the classifier based on FinLex with no additional manual curation delivers performance that is slightly inferior to that of LM. Therefore, the algorithmically curated financial lexicons do not do as well on these classification tasks, but are close, suggesting that the approach may be a good starting point for developing more word lists for other applications. And, given the slight underperformance relative to LM, it may be good to further hand-curate the initial list prepared algorithmically. We note that we used just 4 features (positivity, negativity, uncertainty, and litigiousness) for both LM and FinLex. Third, for the last dataset (Reddit), FinLex outperforms the other methods. Fourth, we added the new lists we curated algorithmically with no further manual curation (for fraud, negligence, riskiness, and safety), and we call this feature set FinLex+. This feature set includes additional word lists created by the algorithm that are likely to contain words that distinguish positive sentiment from negative sentiment, so the improvement we see in classification accuracy is reasonable, and also suggests that these new word lists are an additional contribution.

4 Discussion

We show how pre-trained word embeddings based on FastText may be used to generate financial lexicons for use in classification models. This extends the current popular word lists provided by Loughran and Mcdonald (2011), Loughran and McDonald (2014) that are widely used for financial classification by practitioners and academics. On examination, the generated word lists contain words that are remarkably related to the concepts of interest.

Comparison of LM classifier performance with a TFIDF-based classifier shows comparable performance supporting the use of the LM word lists by practitioners over the past decade. Comparison of the FinLex word lists shows comparable performance to that of the LM word lists but also of sufficient accuracy to suggest that word embedding based lexicons have fruitful practical applications. Using the same algorithm to generate four additional word lists (fraud, negligence, riskiness, safety) further improves model performance.

This work may be extended to other financial applications and datasets rather than just news classification. Examples include credit scoring, scoring of analyst reports, market risk scoring using the new risk and safety scores, and predicting litigation by enhancing the LM litigious word list with the FinLex fraud and negligence list, using the lexicon algorithm on other datasets. Whereas news articles are short and maybe more amenable to numeric scoring as undertaken in this paper, it may be more interesting to compare LM with FinLex using larger financial documents such as regulatory filings, legal dockets, and analyst earnings calls. As the availability of labeled datasets for these tasks becomes more widely available, developing new word lists using the algorithm in this paper will shorten the time to industrial application. Other fruitful topics for further research include using word embeddings trained on finance related datasets as well as application of our algorithm to generate word lists for other industries.

References

Werner Antweiler and Murray Z. Frank. 2004. Is All That Talk Just Noise? The Information Content of Internet Stock Message Boards. *The Journal of Finance*, 59(3):1259–1294. Publisher: [American Finance Association, Wiley].

Dogu Araci. 2019. FinBERT: Financial Sentiment Analysis with Pre-trained Language Models. *arXiv:1908.10063 [cs]*. ArXiv: 1908.10063.

Mirjana Pejic Bach, Zivko Krstic, Sanja Seljan, and Lejla Turulja. 2019. Text Mining for Big Data Analysis in Financial Sector: A Literature Review. *Sustainability*, 11(5):1–27. Publisher: MDPI, Open Access Journal.

Eli Bartov, Lucile Faurel, and Partha S. Mohanram. 2017. Can Twitter Help Predict Firm-Level Earnings and Stock Returns? *The Accounting Review*, 93(3):25–57.

Elizabeth Blankespoor, Gregory S. Miller, and Hal D. White. 2014. The Role of Dissemination in Market Liquidity: Evidence from Firms' Use of Twitter™. *The Accounting Review*, 89(1):79–112. Publisher: American Accounting Association.

Andriy Bodnaruk, Tim Loughran, and Bill McDonald. 2015. Using 10-K Text to Gauge Financial Constraints. *Journal of Financial and Quantitative Analysis*, 50(4):623–646. Publisher: Cambridge University Press.

Samuel B. Bonsall, Andrew J. Leone, Brian P. Miller, and Kristina Rennekamp. 2017. A plain English measure of financial reporting readability. *Journal of Accounting and Economics*, 63(2):329–357.

Tatiana Chebonenko, Lifeng Gu, and Dmitriy Muravyev. 2018. Text Sentiment's Ability to Capture Information: Evidence from Earnings Calls. SSRN Scholarly Paper ID 2352524, Social Science Research Network, Rochester, NY.

Cathy Yi-Hsuan Chen, Roméo Després, Li Guo, and Thomas Renault. 2019. What makes cryptocurrencies special? Investor sentiment and return predictability during the bubble. IRTG 1792 Discussion Paper 2019-016, Humboldt University of Berlin, International Research Training Group 1792 "High Dimensional Nonstationary Time Series".

Lauren Cohen, Christopher Malloy, and Quoc Nguyen. 2020. Lazy Prices. *The Journal of Finance*, 75(3):1371–1415. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/jofi.12885.

Lin William Cong, Tengyuan Liang, Baozhong Yang, and Xiao Zhang. 2019a. Analyzing Textual Information at Scale. SSRN Scholarly Paper ID 3449822, Social Science Research Network, Rochester, NY.

Lin William Cong, Tengyuan Liang, and Xiao Zhang. 2019b. Textual Factors: A Scalable, Interpretable, and Data-driven Approach to Analyzing Unstructured Information. SSRN Scholarly Paper ID 3307057, Social Science Research Network, Rochester, NY.

Sanjiv R. Das and Mike Y. Chen. 2007. Yahoo! For Amazon: Sentiment Extraction from Small Talk on the Web. *Management Science*, 53(9):1375–1388. Publisher: INFORMS.

Sanjiv Ranjan Das. 2014. Text and Context: Language Analytics in Finance. *Foundations and Trends® in Finance*, 8(3):145–261.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv:1810.04805 [cs]*. ArXiv: 1810.04805 version: 2.

W. Brooke Elliott, Stephanie M. Grant, and Frank D. Hodge. 2018. Negative News and Investor Trust: The Role of $Firm and #CEO Twitter Use. *Journal of Accounting Research*, 56(5):1483–1519. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/1475-679X.12217.

Nick Erickson, Jonas Mueller, Alexander Shirkov, Hang Zhang, Pedro Larroy, Mu Li, and Alexander Smola. 2020. AutoGluon-Tabular: Robust and Accurate AutoML for Structured Data. *arXiv:2003.06505 [cs, stat]*. ArXiv: 2003.06505.

Mine Ertugrul, Jin Lei, Jiaping Qiu, and Chi Wan. 2017. Annual Report Readability, Tone Ambiguity, and the Cost of Borrowing. *Journal of Financial and Quantitative Analysis*, 52(2):811–836. Publisher: Cambridge University Press.

Matthew Gentzkow, Bryan Kelly, and Matt Taddy. 2019. Text as Data. *Journal of Economic Literature*, 57(3):535–574.

Edouard Grave, Piotr Bojanowski, Prakhar Gupta, Armand Joulin, and Tomas Mikolov. 2018. Learning Word Vectors for 157 Languages. *arXiv:1802.06893 [cs]*. ArXiv: 1802.06893.

Peter Hafez, Marko Kangrga, Jose Guerrero-Colon, Francisco Gomez, and Ricard Matas. 2020a. Capturing Alpha From Your Own Digital Content. Library Catalog: www.ravenpack.com.

Peter Hafez, Ricard Matas, Francisco Gomez, Marko Kangrga, Boris Skorodumov, and Alan Liu. 2020b. RavenPack News Sentiment Data Outperforms During Coronavirus Crisis. Library Catalog: www.ravenpack.com.

Gerard Hoberg and Gordon Phillips. 2016. Text-Based Network Industries and Endogenous Product Differentiation. *Journal of Political Economy*, 124(5):1423–1465.

Armand Joulin, Edouard Grave, Piotr Bojanowski, Matthijs Douze, Hérve Jégou, and Tomas Mikolov. 2016. FastText.zip: Compressing text classification models. *arXiv:1612.03651 [cs]*. ArXiv: 1612.03651.

Colm Kearney and Sha Liu. 2014. Textual sentiment in finance: A survey of methods and models. *International Review of Financial Analysis*, 33:171–185.

Edward Francis Kelly and Philip Stone. 1975. *Computer Recognition of English Word Senses*. North-Holland Publishing Company. Google-Books-ID: cHzAxwEACAAJ.

Feng Li. 2010. Textual analysis of corporate disclosures : a survey of the literature. *Journal of accounting literature*, 29:143–165.

Kai Li, Feng Mai, Rui Shen, and Xinyan Yan. 2020. Measuring Corporate Culture Using Machine Learning. *The Review of Financial Studies*, page hhaa079.

Tim Loughran and Bill Mcdonald. 2011. When Is a Liability Not a Liability? Textual Analysis, Dictionaries, and 10-Ks. *The Journal of Finance*, 66(1):35–65. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1540-6261.2010.01625.x.

Tim Loughran and Bill McDonald. 2014. Measuring Readability in Financial Disclosures. *The Journal of Finance*, 69(4):1643–1671. Publisher: [American Finance Association, Wiley].

Tim Loughran and Bill Mcdonald. 2016. Textual Analysis in Accounting and Finance: A Survey. *Journal of Accounting Research*, 54(4):1187–1230. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/1475-679X.12123.

Tim Loughran and Bill McDonald. 2020. Textual Analysis in Finance. SSRN Scholarly Paper ID 3470272, Social Science Research Network, Rochester, NY.

Pekka Malo, Ankur Sinha, Pekka Korhonen, Jyrki Wallenius, and Pyry Takala. 2014. Good debt or bad debt: Detecting semantic orientations in economic texts. *Journal of the Association for Information Science and Technology*, 65(4):782–796.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. *arXiv:1301.3781 [cs]*. ArXiv: 1301.3781.

Tomas Mikolov, Edouard Grave, Piotr Bojanowski, Christian Puhrsch, and Armand Joulin. 2017. Advances in Pre-Training Distributed Word Representations. *arXiv:1712.09405 [cs]*. ArXiv: 1712.09405.

Bryan R. Routledge. 2019. Machine learning and asset allocation. *Financial Management*, 48(4):1069–1094. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/fima.12303.

Philip Stone, Dexter Dunphy, Marshall Smith, and Daniel Ogilvie. 1966. *The General Inquirer: A Computer Approach to Content Analysis*. The MIT Press.

FinLex: An Effective Use of Word Embeddings for Financial Lexicon Generation

Authors:

1. Sanjiv R. Das, Santa Clara University and AWS, srdas@scu.edu
2. Michele Donini, AWS (Amazon Web Services), donini@amazon.com
3. Muhammad Bilal Zafar, AWS, zafamuh@amazon.com
4. John He, AWS, hezhijia@amazon.com
5. Krishnaram Kenthapadi, AWS, kenthk@amazon.com

Abstract: We present a simple and effective methodology for the generation of lexicons (word lists) that may be used in natural language scoring applications. In particular, in the finance industry, word lists have become ubiquitous for sentiment scoring. These have been derived from dictionaries such as the Harvard Inquirer and require manual curation. Here, we present an automated approach to the curation of lexicons, which makes automatic preparation of any word list immediate. We show that our automated word lists deliver comparable performance to traditional lexicons on machine learning classification tasks. This new approach will enable finance academics and practitioners to create and deploy new word lists in addition to the few traditional ones in a facile manner.

Keywords: Lexicons, embeddings, scoring, machine learning