A machine learning approach to domain specific dictionary generation. An economic time series framework

HANJO ODENDAAL

Stellenbosch Economic Working Papers: WP06/2021

www.ekon.sun.ac.za/wpapers/2021/wp062021

March 2021

KEYWORDS: Sentometrics, Machine learning, Domain-specific dictionaries
JEL: C32, C45, C53, C55

DEPARTMENT OF ECONOMICS
UNIVERSITY OF STELLENBOSCH

SOUTH AFRICA

A WORKING PAPER OF THE DEPARTMENT OF ECONOMICS AND THE
BUREAU FOR ECONOMIC RESEARCH AT THE UNIVERSITY OF STELLENBOSCH

www.ekon.sun.ac.za/wpapers

A machine learning approach to domain specific dictionary generation. An economic time series framework

Hanjo Odendaal[a,*]

[a]*Department of Economics, University of Stellenbosch, South Africa*

Abstract

This paper aims to offer an alternative to the manually labour intensive process of constructing a domain specific lexicon or dictionary through the operationalization of subjective information processing. This paper builds on current empirical literature by (a) constructing a domain specific dictionary for various economic confidence indices, (b) introducing a novel weighting schema of text tokens that account for time dependence; and (c) operationalising subjective information processing of text data using machine learning. The results show that sentiment indices constructed from machine generated dictionaries have a better fit with multiple indicators of economic activity than Loughran and McDonald (2011)'s manually constructed dictionary. Analysis shows a lower RMSE for the domain specific dictionaries in a five year holdout sample period from 2012 to 2017. The results also justify the time series weighting design used to overcome the $p >> n$ problem, commonly found when working with economic time series and text data.

Keywords: Sentometrics; Machine learning; Domain-specific dictionaries

JEL classification C32; C45; C53; C55

1. Introduction

This paper aims to offer an alternative to the the manually labour intensive process of constructing a domain specific lexicon or dictionary through the operationalization of subjective information processing. Traditionally, in order to analyze the tone of any news texts, computational linguistics

*Corresponding Author

**Acknowledgements: This research was financially supported by the Bureau for Economic Research at Stellenbosch University and 71Point4, a Cape Town based strategic research consultancy. We would also like to thank the team at News24 and Sabinet for providing data that made this research possible

Email address: hanjo.oden@gmail.com (Hanjo Odendaal)

employs manually pre-selected dictionaries. These dictionaries range from general (Harvard-IV, Liu (2012) and Nielsen (2011)) to more domain specific lexicons such as Loughran and McDonald (2011) and Henry (2008) which have a financial focus. However, there remains substantial differences in the word lists of theses domain specific dictionaries. Despite the range of dictionaries to choose from, one dictionary alone might not be sufficient to capture the nuances found within a specific news domain. The choice of a suitable dictionary poses a challenge in itself, before any analysis can begin.

Machine learning is presented as one way to create such a domain-specific dictionary. This approach would enable the creation of dictionaries tailored to a specific need in an automated fashion. Being less subjective, these dictionaries are also more easily tested and replicable. Tokens (words) are statistically selected using recursive feature elimination to generate a domain-specific dictionary from a corpus of text. An individual domain-specific sentiment lexicon and respective sentiment index is generated for the Consumer confidence Index (CCI), Business Confidence Index (BCI), Building Confidence Index (Building), Civil Confidence Index (Civil), Purchasing Managers Index (PMI), as well as the leading indicator constructed by the South African Reserve Bank (Leading). To accomplish this, we applied a random forest algorithm to select the most important words from a corpus of business news media, determined whether the token has a positive or negative relationship with the outcome through ordinary least squares, and constructed a corresponding index.

Building on current empirical literature that has explored the media-economic nexus, we contribute by (a) constructing a domain-specific dictionary for various economic confidence indices, (b) introducing a novel weighting schema of text tokens that aim to account for time dependence, and (c) operationalising subjective information processing of text data using machine learning.

The machine-generated dictionaries contain both unigrams and bigrams as features, creating a richer feature space in comparison to unigram tokens. Sentiment indices constructed from machine-generated dictionaries are shown to have a better fit with the multiple indicators investigated when compared to the sentiment index constructed from a commonly used financial dictionary. The domain-specific sentiment indices also show a significantly lower root mean squared error (RMSE) in the five-year holdout sample period from 2012 to 2017. These results support the case for domain-specific dictionaries being able to pick up nuances found within domain-specific topic news. The results also suggest that having a manually generated dictionary act as a prior narrows the tokens that the Random Forest has to search over, while maintaining the same lower RMSE out-of-sample as an unrestricted model that includes all the tokens. Employing a manually generated

dictionary such as that of Loughran and McDonald (2011) also decreases the computation burden on the pre-processing[1] and estimation of the dictionaries.

This paper starts off by examining the relationship between expectations and the role it plays in shaping how the economy evolves. Section 2 highlights the reason why expectations matter by illustrating that expectations are able to provide foresight and as such play an important part in policy design. The section draws from the inflation expectation literature to provide a theoretical basis for how the media forms expectations which, in turn, influence the real economy. We also expand how computational linguistics is applied within economics to derive economic sentiment, especially utilising text from media sources. The section ends by providing an overview of the current methods and techniques that are utilised to create domain-specific dictionaries. Following an extensive overview of the theoretical arguments and relevant literature, section 4 discusses the framework to create domain-specific dictionaries. This section focuses on the random forest algorithm that is used, as well as the subjective choices that have to be made in order to generate a domain-specific dictionary using the proposed methods. Section 5 follows to provide a thorough evaluation of the domain-specific dictionaries and the respective sentiment indices created from them. The dictionaries are compared in terms of the tokens each of the dictionaries consists of. We also showcase the specific bigrams that were selected by the algorithm as a network analysis. After the dictionaries' constructs are discussed, each of their resulting sentiment indices is evaluated against various confidence indices using RMSE as a measure to determine the best fit. Lastly, section 6 ends with the conclusion and suggestions for future research.

2. The news, the economy, and text analysis

2.1. The news and economy

The study of the closely related relationship between economic reporting and the real economy is not new in terms of scholarly attention. Early work such as that of Goidel and Langley (1995) already identified a link between news coverage and how it influences both the perception and evaluation of the real economy. Goidel and Langley (1995) were also some of the first researchers to posit that the media tends to follow negative economic conditions more closely and is accordingly

[1] Pre-processing is that step in which the data gets transformed, or encoded, to bring it to such a state that the machine can easily parse the features of the data.

characterised by a set of persistent biases. Given this known bias, it is important to also acknowledge that the media exercises plenty of latitude in what they deem to be important. This entails that the relationship between the economy and economic media is not stable, but dynamic. This dynamic relationship between the economic agent and news media can also be seen in the work of Carroll (2003). The author illustrate that although empirical household expectations cannot be deemed to be rational, expectation dynamics can be explained through a model that incorporates professional forecasters' views. In this model, the household's views and expectations are influenced by news reports which reflect on the views of professional forecasters, who, in turn, could be considered to be rational agents. The agents absorb macroeconomic trends and economic content by using a model of probabilistic absorption of news stories as opposed to assimilating raw official statistics. This inattention to news stories by choice creates a stickiness in aggregate expectations that, in turn, have important macroeconomic consequences. The baseline model found in Carroll (2003) follows a very similar approach to Mankiw and Reis (2002) in modelling the evolution of mean expectations. Apart from finding that expectations are sticky due to rational inattention (see Sims (2003)), the author also find that the data showed a preference for a forward-looking version of the model. This is in contradiction with an adaptive expectations model whereby expectations are adjusted in line with recently reported statistics. The results of the paper are important as they act as a theoretical model in explaining and motivating that news reports impact forward-looking expectations. This model provides a plausible middle ground between fully rational expectations and adaptive expectations.

The stickiness of expectations is further explored in a labour market model proposed by Akerlof et al. (2000). Here, the economic agents only concern themselves with inflation when the ignorance thereof will become costly. This rational inattention model can also be applied in the context of news sentiment as discussed further below. In the inflation framework, when inflation is low, it is not very salient and the relationship between expected inflation and the wage and price setting is not as strong. But as inflation rises, so does the importance of anticipating it correctly, resulting in the price and wage setting fully responding to inflation expectations. Given that the largest expense for most businesses would be wages, if the household's expectations of inflation translate to a change in nominal wage demand, the firm's pricing decisions will be affected through the usual wage-push channel. In much the same way as inflation plays a larger role in the price-wage setting as it increases, so does the tone and increased volume of reporting of the economy (in the news)

become important to economic agents as it bears direct cost to ignore.

Lamla and Lein (2014) examine the role of news media and how it influences consumer expectations around inflation. They emphasise the role of information rigidities and how the media plays an important part in transmitting information about the macroeconomic conditions to the consumer. Through the use of a theoretical model, the authors show that the media affects consumers and expectations mainly through two channels: volume and bias. The higher the volume of news available to the agents in the economy, the more accurate their expectations become; but this effect could be reversed if the news is consistently biased. Using a dataset of German news articles, the results show that given a higher volume of information, agents have a higher propensity to update their expectation, resulting in more accurate forecasts. The empirical results also provide evidence that a one standard deviation increase in neutrally toned media improved consumer's expectations by around 20%, while the increase of negatively toned media deteriorated the results by roughly the same amount. This direct effect on consumer's expectations also has an indirect effect on the real economy. Inflation expectations have a tendency to be self-fulfilling (Leduc et al., 2007). As inflation expectations keep adjusting upwards, the effectiveness of monetary policy could be impeded. This interplay between the flow of information through the news media and the real economy is not only observed in the adjustment of inflation expectations, but also in the adjustment of broader macroeconomic expectations.

Nadeau et al. (1999) evaluated the process by which business elites' expectations and retrospectations trickle through to news media, eventually impacting not only the economy but also matters such as presidential approval. By using content analysis, the results showed that although information is transmitted through news media channels, it is not always completely unchanged. The media is partially autonomous, acting as a mediator between expert opinion and the mass public on economic conditions. The author's research builds on the idea of media-dependency theory in which the experts' economic views will be widely and correctly reported by journalists. For the mass public, this reported information is by far the best information on the future state of the economy. But, with economic reporting overemphasising poor economic news, the public does not always adjust their expectations contemporaneously due to rational inattention, as it is a costly exercise to do so.

Although research such as that of Carroll (2003), Akerlof et al. (2000), and Lamla and Lein (2014) focuses on how the media can influence expectations around the inflation that eventually

flows through to the real economy, Nadeau et al. (1999) demonstrate how general economic coverage by the news media can subsequently affect public opinion about any economic or political outcome in a similar manner. This in part questions the extent of the influence of news media and if, in a rational inattention model, the news says something about the past, present, or future of the economy. Soroka et al. (2015) explored this question empirically.

They asked the question of whether media news coverage has a forward-looking or retrospective agenda towards economic sentiment. To test this, Soroka et al. (2015) relied on three sources of data: macroeconomic measures, media data, and measures of public opinion. The data spans the period 1980 to 2011. The macroeconomic data contains the leading, coincidental, and lagging indicator of the economy, which allows for the assessment of the impact of news media on each of the indicators. The indicators are purged from all sentiment measures so as not to introduce measure-induced endogeneity. The media data consisted of articles from the New York Times and Washington Post relating to major economic issues and had been transformed into indices indicating tone (using sentiment analysis) and volume of reporting. To test the relationship between the media and economy, error correction models were used, as they have the advantage of breaking down the long-term and short-term dynamics between series. Results from the model showed that the tone and volume of media reporting were significantly related to present and future economic activity, while the relationship with lagging indicators was insignificant. When the indicators were estimated in a saturated model (all indicators included), the leading indicator was the only significant variable. This indicates that conditioning on other indicators in the economy (ceteris paribus), the media was more concerned with reporting on the future.

Contemporary literature extends the discussion around the media-economic connection by analysing text through what is known as text analysis or computational linguistics. In recent years, computational power has become more accessible, but at the same time the amount of information has increased drastically. This upsurge in information content can mainly be accredited to two large drivers: social media and online news media. Both of these factors have led to new avenues of research being developed to gain a better understanding on the role that the media plays in shaping expectations and the real economy. A large portion of text analysis within economics focuses on quantifying the sentiment of economic agents. This could be anything from deriving consumer and business sentiment through analysing news and blogs to investigating official central bank releases for information on macroeconomic policy direction.

Given the growing application of sentiment analysis to answer various economic questions, it is imperative that the right sentiment dictionary be used to analyse the text. Using machine learning algorithms to build domain-specific dictionaries could provide a possible avenue to create these dictionaries more efficiently in an automated fashion.

3. Machine learning to build domain-specific dictionaries

One of the earliest works on automatic dictionary generation involves polarising tokens, assuming words act the same as the spin of an electron. The model of Takamura et al. (2005) extracts semantic orientation by comparing the directional spins of an electron (up or down) to the semantic orientation of words (positive or negative). Using the Spin Model and applying the mean field approximation, the authors compute the average orientation of each word. Using this method approximates the probability function of the system instead of computing the intractable actual probability function. This allows for the incorporation of more noisy data in the analysis. This was not previously possible with methods that focused on bootstrapping and shortest-path techniques to calculate semantic orientation. The proposed Spin Model outperformed both the bootstrap and shortest path in out-of-sample tests in terms of percentage precision. In more recent literature, text mining has started to apply machine learning methods to automatically construct a dictionary for sentiment analysis.

To better understand how companies frame their press releases, Pröllochs et al. (2015) constructed domain-specific dictionaries using announcement data. To do this, the authors used three different Bayesian approaches, namely LASSO, Ridge, and elastic net regression. These methods have been shown to have high explanatory power while also allowing for inference. All three of the models rely on Bayesian regularisation to conduct variable selection that shrinks the coefficients of non-informative variables. In the case of generating a dictionary, the model will converge to a parsimonious selection of tokens with high explanatory power. This approach permits for weights to be calculated on how strongly investors are influenced by information in the form of selected words. The authors do highlight that tokens identified as positive (negative) by the model, may not necessarily be interpreted positively by investors. The information contained in words is highly dependent on the context in which they appear. This means that one cannot assume that the model or manually generated polarity of a word necessarily equates to the linguistic orientation of the word (Loughran and McDonald, 2011). The generated dictionaries are compared to existing financial

dictionaries by evaluating their predictive performance for sentiment analysis on a validation set. Unlike previous methods, the announcements are not manually labelled; Pröllochs et al. (2015) instead use the corresponding stock market returns as the objective measure. The results showed that the ridge regression had the highest predictive performance out-of-sample, with an improvement of a 93.25% increase in correlation in comparison to other well-known financial dictionaries.[2]

Machine learning has also been applied to forecasting key economic variables using newspapers. Using the text from three different newspapers, Kalamara et al. (2020) use a novel method of text counts, similar to the one used in this paper, to extract timely signals to forecast GDP, CPI and unemployment. The findings showed that by using a supervised method that identifies relevant tokens, the method employed improved the forecasts when compared to existing text-based methods. Furthermore, the improvements were most prominent in periods of stress, where accurate forecasts are vital.

In contrast with the machine learning approach found in Pröllochs et al. (2015) and Kalamara et al. (2020), Labille et al. (2017) used probabilistic and information theory techniques to construct a domain-specific dictionary. By moving away from transferred supervised machine learning techniques, the method has the advantage of not having to update or adapt the constructed dictionary. To generate the three dictionaries, the authors used Amazon product reviews for 15 different categories submitted from January 2013 to July 2014. The reviews were star-rated 1 to 5. For their experiment, 1- and 2-star ratings were deemed negative, while 4- to 5-star reviews were labeled positive. This probabilistic approach was shown to outperform generic dictionaries with higher accuracy and F1-scores, indicating that for their experiment, domain-specific dictionaries were more accurate in the task of sentiment analysis.

4. Creating a domain-specific dictionary using random forests

This section provides an overview of the research methodology these authors implemented to generate domain-specific dictionaries using random forests.

The Random forests (RF) technique is increasingly used in a range of fields due to its high prediction ability and the feature selection procedure it inherently contains (Athey et al., 2016; Biau

[2]The correlation coefficient, ρ, was 0.1030 for the Ridge regression, while the Harvard-IV dictionary performed the best among the manually generated dictionaries with $\rho = 0.0533$.

and D'Elia, 2009; Meinshausen, 2006). Although RFs have been widely applied in bioinformatics and related fields, very few applications within economics have seen the use of this algorithm in time-series problems. This is mainly due to the fact that the algorithm assumes each data point to be independent and so violates the time dependency assumption within time-series data. Despite the algorithm's flaws in terms of pure prediction or forecasting in time-series problems, RFs can still be used as a feature selection machine where the curse of dimensionality, $p >> n$, is a problem (Kane et al., 2014; Tyralis and Papacharalampous, 2017). Using a machine learning model such as LASSO does not have this advantage and accordingly the number of tokens selected from the feature space (p) will always be restricted to the number of observations (n).[3]

The section is broken up in three subsections: how the token design matrix was constructed, explanation of the recursive feature elimination technique used to build the dictionaries and then an overview of practical considerations when using Random Forests is discussed.

4.1. Token weights

Before applying the random forest model, pre-processing steps were applied to the text. As is customary in any text analysis, the first step was to tokenise the text so that the columns present a count of a token for a given t in time. Next, common words, often called stop words (including conjunctions), were removed from the text as these words contain little to no information value.

Once this procedure had been completed, a time-series weighting schema was applied to the design matrix of tokens. We weigh the tokens both within and over time in the design matrix.[4] This weighting assumes that the volume of a given term acts as a signal to agents to become more rational, increasing absorption of the specific economic term, thereby adjusting their expectations closer to the actual outcome. Assume that the raw count for a given term is defined as $f_{t,j}$ and the frequency, $f_{t,j}^{freq} = f_{t,j} / \sum_{j' \in t} f_{t,j'}$, where t is a corpus of text in time and j the token. The relative frequency of the term across time is then normalised by dividing it by the maximum frequency for a given training period T_{n-m}:

[3] An example of this would be if the time series consisted of quarterly data for 10 years. If LASSO was used, the maximum number of tokens that the algorithm would select as significant would be 40 (10 years × 4 quarters).

[4] This is different from the usually applied term-frequency, inverse document frequency (tf-idf) transformation that looks to highlight importance of word in a single document, in that the paper's transformation looks to highlight the importance across documents (in this case, time).

$$f_{t,j}^{rel} = \frac{f_{t,j}^{freq}}{\max\{j`, \in T_{n-m}\} f_{t,j'}^{freq}} \tag{1}$$

where m is specified so as to divide the time series corpus into a training (in-sample) and test (out-of-sample) set.[5] Setting m so that the data is split on January 2012, the training sample contained 41 observations, while our out-of-sample period contained 21 quarters. The same transformation needed to be applied to the test portion of the data, T_{n+m} using the maximum frequency from the test period as the denominator. This was to ensure that no information leakage occurred between the training and test periods.

The final step before the selection procedure could be carried out was to apply near-zero variance (NZV) analysis to the features. In many cases, having predictor variables with low cardinality (also known as zero variance) will decrease model performance. This is most evident when estimating linear regressions where numerical problems occur if any of the estimators are near-zero variance predictors. In the case of text analysis, this problem is exacerbated, as some tokens might only occur in one quarter of the text sample. Given that the aim of the paper is to identify a domain-specific dictionary, if NZV predictors were included, the RF model could overfit using these single occurrence tokens with a specific level of the outcome variable. This would result in the algorithm including words in the final dictionary that have no relevance to the outcome or, in our case, absolutely no economic connotation. Identifying and removing NZV predictors also plays a significant role in reducing the feature space of tokens. When working with bi- and unigram tokens, the feature space explodes into the millions, while most of the features have no informational value. To identify predictors with NZV characteristics, two properties can be examined (Kuhn and Johnson, 2013). The first property takes into account the percentage of unique values. The higher the percentage of unique values, the less information value the predictor has, and it will not generalise well when we apply the dictionary out of sample. There is no correct value or proportion which optimises the distinction between NZV and non-NZV. We used an arbitrary unique value cut-off of 20 as it corresponds to a token needing to be unique in half of the training sample period. The second criterion on which a token is examined is the skewness of the frequency distribution of the variable.

[5]The choice of m will influence the weighting outcome as the token weight remains anchored to its max within the training sample.

If the most frequent occurrence is a much larger factor of the second most frequent token, the predictor might be highly skewed and the frequency of the variable imbalanced. Kuhn and Johnson (2013) suggests that if the most frequent value is a factor of 20 of the second most frequent value, it should be discarded. This is not a feasible number as we are working with quarterly data and a training sample of only 41 observations. We decided to apply a much stricter filtering rule of a frequency cut-off factor of 1.5.

Both of these criteria are used together to flag variables that could be potential NZV predictors. After applying this filtering process, the number of possible tokens for the dictionary dropped significantly from 14.6 million tokens to 55 000. This step could potentially be used to further scale down the number of tokens under consideration in order to lessen the computational burden, although it was not further researched.

4.2. Feature selection through RFE

Having filtered out the tokens with little to no informational value, the feature set still contained 55 000 tokens from which to generate a domain-specific dictionary. From a practical viewpoint, a dictionary with very few tokens that is able to capture the underlying trend in the economic indicator is beneficial due to its ease of dissection. If a dictionary only has a few hundred words, it is easier to understand what is causing the shift in the constructed sentiment index. In terms of a statistical property, fewer tokens introduce less noise (or complexity) into the modelling process, which could, in turn, negatively affect the desired outcome of a good fit. Certain models such as tree- and rule-based models have a natural resistance to non-informative predictors purely due to their mathematical constructs.

To select the final tokens, we employed a wrapper method known as recursive feature elimination (RFE) described in Guyon and Elisseeff (2003). This backwards selection algorithm is the main procedure that is used to create the domain-specific dictionary. It starts off by estimating a model over the whole feature space, ranking the importance of each variable by some measure. In the case of Random Forests, this is done through importance criteria such as impurity, corrected impurity, or permutation. Once the initial fit has been estimated, only the S_i most important variables (tokens) are kept and a new model is estimated from these remaining predictors. This process is continued for the specified subset of predictor variables specified by the user over either resampling or time-slices

as in the case of time series.

Algorithm 1: Backward selection via the recursive feature elimination algorithm

```
for each timeslice iteration do
    Partition data into training and hold out set via timeslices;
    Tune/train the model on the training set using all P predictors;
    Calculate model performance;
    Calculate variable importance;
    for each subset size S_i,   i = 1, ... S do
        Keep the S_i most important variables;
        Tune/train the model on S_i predictors;
        Calculate model performance;
        Recalculate variable importance;
    Calculate the performance profile over the S_i using the held-back sample;
    Determine the appropriate number of predictors (Set of S_i associated with best
     performance);
    Fit the final model based on the optimal S_i;
```

where this paper selected $S = \{100, 150, 200, \ldots, 3500\}$. This results in the domain-specific dictionaries containing between 100 and 3500 tokens (multiples of 50). This selection was done as searching over the whole feature space would be very time consuming.

The chosen model used in steps of 3, 8, and 14 of the RFE algorithm (1) is the RANdom forest GEneRator (Ranger) algorithm described in Wright and Ziegler (2015). The Ranger RF algorithm is a highly optimised Random Forest implementation in C++ with a focus on the analysis of highly dimensional data.

When we use RFs for regression, the procedure is making use of bootstrap aggregation or bagging. This fits the regression tree to many bootstrap-sampled versions of the training data and aggregates the estimations for the final result. Trees are excellent candidate algorithms for the concept of bagging as they have the ability to capture complex interactions in the structures of the data, while having relatively low bias if grown significantly deep. The general Random Forest algorithm

described in Breiman (2001) can be characterised as follows:[6]

Algorithm 2: Random forest for regression

for $b = 1$ *to* B **do**

  Draw bootstrap sample $\mathbf{Z}^*$ of size N from training data;

  Grow a RF tree T_b to the bootstrapped data by recursively repeating following steps for each terminal node of the tree until minimum node size n_{min} is reached;

  while $n_{min}! = min$ **do**

    Select m variables at random from p variables;

    Pick the best variable to split among m;

    Split node into 2 daughter nodes;

Output ensemble of tree ${T_b}_1^B$ Make prediction $\hat{f}_{rf}^B(x) = \frac{1}{B}\sum_{b=1}^{B} T_b(x)$

The choice of variable importance in algorithm (1) plays a significant role in which variables are kept in S_i. Although several variants of importance criteria exist, it has been shown that by using the well known Gini impurity variable importance measure, the selection of features are biased towards features with more categories or continuous variables (Strobl et al., 2007). Given the known bias of the aforementioned, for the RFE procedure we use the permutation accuracy importance measure in its estimation step. The procedure selects a strong predictor, X_i, of the outcome and randomly permutes it. This results in its original relationship with the outcome variable no longer being retained. The permuted variable is then used in combination with the remaining unpermutated variables to predict Y. If the prediction accuracy decreases substantially, the original X_i can be deemed to be an important variable. Besides overcoming the bias associated with the univariate screening methods, the permutation test not only tests each predictor individually, but also takes into account multivariate interactions with the other predictor variables.

Next, the polarity of a given word is estimated. Once the final set of tokens has been selected as part of the RFE procedure, a simple linear model is used:

$$Y = \beta X_i + \varepsilon \tag{2}$$

where Y is the outcome variable and X_i is the remaining token $i \in S_i$. If the resulting coefficient

[6] For an extensive overview of tree-based methods see Friedman et al. (2001).

β_i is greater than zero, the variable is deemed to have a positive relationship and vice versa. This estimation occurs for all $\mathbf{X}$ resulting in a final word list.

The whole algorithm from start to finish can succinctly be described as:

Algorithm 3: Creating domain-specific dictionary and sentiment index

Tokenise time-series text into n-grams;
Convert into tokens into document feature matrix T;
Apply time-series weighting to train design matrix as in eq (1);
Apply weighting to out of sample;
Use near-zero variance for feature elimination;
Select X_i features using RFE procedure for training data T;
for *each selected* X_i **do**
    Fit linear regression $Y = \beta X_i + \varepsilon$;
    if $\beta > 0$ **then**
        Polarity = Pos_i;
    else
    if $\beta < 0$ **then**
        Polarity = Neg_i;
    else
        Polarity = $Neutral_i$;
Construct sentiment index $I_{it} = \frac{\sum Pos_{it} - \sum Neg_{it}}{(\sum Pos_{it} + \sum Neg_{it})}$ $\forall$ T from $\mathbf{X}$;

4.3. Practical considerations

In creating a domain-specific dictionary and its respective sentiment index, several different specifications were evaluated. The experimental design is aimed at evaluating four different levers that any researcher will be faced with when applying the procedure described in algorithm (3). This resulted in having to estimate 16 different models for each of the indicators.

The first adjustable lever in automatic dictionary creation is the restriction placed to determine which tokens are eligible. Initially, the procedure is run without any restrictions on the 55 000 tokens remaining after the NZV filtering. The resulting model will be referred to as the *unrestricted* model for the remainder of the paper. Afterwards, we placed a restriction on the words by saying

that a token needs to be in the Loughran dictionary: $(W_i) \in W_L$ where W_i represents a token. For unigram tokens, this is straightforward, but for the bigrams, a different approach is considered: $(W_{i1} \in W_L)$ OR $(W_{i2} \in W_L)$. The bigram is broken up into two separate unigrams and if either of them is found in the Loughran dictionary, then the bigram remains as a token. Thus, the Loughran dictionary is used as a prior in the generating process.[7] This restricted specification is referred to as the *Loughran prior* model when the results are discussed. By applying this restriction on the tokens, the feature space decreases from 55 000 to 5600.

The second choice the user has to make is the number of trees to use in the estimation step. The general rule is that more trees are required for stable variable importance estimates, but it does increase the estimation and prediction time. To test the influence of the number of trees selected for the procedure, we run both a 500- and 1000-tree variant of the model.

Next, the tuning parameter that decides on the number of variables that can be selected at each split of the tree node needs to be given: *mtry*[8]. In a study conducted by Genuer et al. (2010), the authors examined what influence the *mtry* parameter had on the variable importance measure. They concluded that using a large mtry leads to much higher magnitudes on the variable importance measure, which, if using the Gini method, could bias the outcome. Grömping (2009) found, in turn, that a high *mtry* parameter gives lower importance to weak predictor variables in high-dimensional datasets, especially when the trees are deep. If left to the default, very shallow trees will be built and the importance of each token will be almost equally distributed. The influential impact of the *mtry* tuning parameter was tested using the following specification: high $= \sqrt{p}$ and low $= \sqrt{p}/3$, where p is the number of tokens considered. The default recommendation for *mtry* in a regression problem is $\sqrt{p}/3$, but given the high-dimensional nature of text tokens, it is possible that having a higher *mtry* could lead to better filtering of important tokens in the variable importance step of the procedure. To achieve an even deeper tree, the minimum node size was decreased. The minimum node size was set to 1.

The last consideration taken into account is the time-dependent nature of the outcome variable. To account for the autocorrelation inherent in time-series data, a separate specification was introduced where the design matrix included the previous period's outcome in levels, as well as the percentage

[7] The use of the word prior is not to be confused with the prior used in Bayesian methods, although it acts in a similar way to giving the algorithm a starting point for the dictionary creation.

[8] Number of variables randomly sampled as candidates at each split.

change between periods. This means that the model had to its availability the choice of conditioning its selection of tokens on the time-series characteristic of the outcome variable.

5. Evaluation of domain-specific dictionaries and sentiment indices

This section starts by discussing and comparing the different dictionaries generated using the procedure presented in algorithm (3). We then proceed to evaluate the sentiment indices constructed using the generated dictionaries for each of the six important sentiment indicators produced by the BER and the SARB namely: Business Confidence (BCI), Building Confidence (Build), Consumer Confidence (CCI), Civil Confidence (Civil), Purchasing Manager's Index (PMI) and the SARB Leading Business Cycle indicator (Leading).

5.1. Generated dictionaries

Table 1 shows all of the model specifications, the number of tokens, and the proportion of positive and negative words for each resulting dictionary. We can see that the unrestricted models generated much larger dictionaries for the BCI, Building and Civil Confidence indices, while for CCI and PMI, all specifications generated large dictionaries.[9] This would suggest that the models that performed the best in terms of cross-validation were more complex models consisting of a large number of tokens as predictors. Another observation from the table is the low number of tokens selected when we used the Loughran dictionary as a prior. A large proportion of these dictionaries only contain 100 tokens, the minimum set as specified in the RFE procedure.

[9] All generated dictionaries are available at https://machne-generated-dict-2020.s3.us-east-2.amazonaws.com/dictionaries.csv.

Table 1: List of specifications under evaluation and the number of tokens that were selected by the RFE procedure. The number in the bracket shows the proportion positive and negative words (positive, negative).

Specification	Type	Inc lag	Tree size	Mtry size	Bci	Build	Cci
A	loughran	FALSE	500.00	high	100 (27%,72%)	950 (31%,68%)	3500 (29%,69%)
A	unrestricted	FALSE	500.00	high	3300 (43%,56%)	1800 (46.3%,51.8%)	3500 (49.40%,50.14%)
B	loughran	TRUE	500.00	high	100 (25%,73%)	200 (30%,67%)	3400 (29%,69%)
B	unrestricted	TRUE	500.00	high	2850 (44%,56%)	3100 (44.9%,53.9%)	3350 (49.16%,49.97%)
C	loughran	FALSE	1000.00	high	100 (26%,73%)	200 (29%,68%)	3450 (29%,69%)
C	unrestricted	FALSE	1000.00	high	3000 (42%,57%)	3300 (43%,56%)	3250 (49.75%,49.63%)
D	loughran	TRUE	1000.00	high	100 (26%,72%)	100 (27%,70%)	3250 (30%,69%)
D	unrestricted	TRUE	1000.00	high	2100 (43%,56%)	3300 (44%,55%)	3400 (49.82%,49.41%)
E	loughran	FALSE	500.00	low	100 (25%,73%)	100 (25%,72%)	2500 (32%,66%)
E	unrestricted	FALSE	500.00	low	2550 (44%,55%)	2550 (43%,55%)	3400 (51.4%,48.1%)
F	loughran	TRUE	500.00	low	100 (22%,76%)	100 (24%,72%)	1550 (36%,63%)
F	unrestricted	TRUE	500.00	low	3050 (44%,55%)	2100 (45.6%,52.9%)	2700 (53.3%,46.0%)
G	loughran	FALSE	1000.00	low	100 (21%,78%)	100 (26%,71%)	3250 (30%,68%)
G	unrestricted	FALSE	1000.00	low	1750 (42%,56%)	2200 (45.9%,52.3%)	2750 (54.5%,44.9%)
H	loughran	TRUE	1000.00	low	100 (23%,75%)	100 (25%,72%)	3100 (31%,67%)
H	unrestricted	TRUE	1000.00	low	3250 (44%,55%)	2650 (44.8%,53.5%)	3500 (52.7%,46.7%)

Specification	Type	Inc lag	Tree size	Mtry size	Civil	Leading	Pmi
A	loughran	FALSE	500.00	high	100 (34%,60%)	550 (43%,57%)	3050 (49.77%,48.79%)
A	unrestricted	FALSE	500.00	high	3250 (47.4%,51.0%)	100 (76%,23%)	3400 (35%,64%)
B	loughran	TRUE	500.00	high	100 (34%,59%)	150 (53.3%,46.0%)	3450 (52.0%,46.3%)
B	unrestricted	TRUE	500.00	high	2700 (47.5%,50.8%)	100 (74%,24%)	3300 (34%,65%)
C	loughran	FALSE	1000.00	high	100 (33%,62%)	150 (54.0%,46.0%)	3450 (51.3%,47.0%)
C	unrestricted	FALSE	1000.00	high	3250 (45.6%,52.6%)	150 (74%,25%)	3350 (33%,65%)
D	loughran	TRUE	1000.00	high	100 (33%,62%)	100 (57%,42%)	800 (25%,74%)
D	unrestricted	TRUE	1000.00	high	3450 (44.9%,53.3%)	100 (74%,24%)	2750 (30%,68%)
E	loughran	FALSE	500.00	low	100 (34%,61%)	3450 (31%,67%)	2250 (43%,56%)
E	unrestricted	FALSE	500.00	low	3200 (47.9%,50.7%)	150 (71%,27%)	1350 (24%,74%)
F	loughran	TRUE	500.00	low	100 (34%,58%)	100 (54.0%,45.0%)	350 (11.4%,86.3%)
F	unrestricted	TRUE	500.00	low	3500 (46.7%,52.2%)	150 (75%,23%)	1500 (26%,73%)
G	loughran	FALSE	1000.00	low	100 (33%,61%)	250 (50.4%,49.2%)	1850 (37%,62%)
G	unrestricted	FALSE	1000.00	low	3350 (46.5%,51.5%)	200 (72%,26%)	3300 (30%,68%)
H	loughran	TRUE	1000.00	low	100 (32%,60%)	100 (53.0%,46.0%)	450 (16%,82%)
H	unrestricted	TRUE	1000.00	low	2550 (47.1%,50.7%)	200 (76%,23%)	3400 (32%,67%)

To measure the overlap for a given outcome and its various dictionaries, we looked at how many of the words occur in all dictionaries. The maximum number of words is bound by the smallest dictionary that is generated. For the Loughran prior specifications, the number of similar tokens among the different dictionaries is quite high. For instance, in the case of the BCI, 74 tokens were similar in eight of the generated dictionaries from different specifications, all of whom only had 100 tokens each.

Table 2: Breakdown of the similarities among the different dictionaries generated from the various specifications.

Sentiment	Type	Bci	Build	Cci	Civil	Leading	Pmi
Distinct Tokens	loughran	127	951	4708	133	3451	4567
Overlapping tokens	loughran	74	82	1346	77	83	330
negative	loughran	53 (72%)	59 (72%)	846 (63%)	45 (58%)	36 (43%)	287 (87%)
neutral	loughran	1 (1%)	2 (2%)	10 (1%)	3 (4%)	0	5 (2%)
positive	loughran	20 (27%)	21 (26%)	490 (36%)	29 (38%)	47 (57%)	38 (12%)
Distinct Tokens	unrestricted	6498	6045	7422	7242	245	7397
Overlapping tokens	unrestricted	748	722	903	870	64	545
negative	unrestricted	402 (53.7%)	359 (49.72%)	371 (41%)	401 (46.1%)	14 (22%)	437 (80%)
neutral	unrestricted	6 (0.8%)	9 (1.25%)	6 (1%)	14 (1.6%)	1 (2%)	6 (1%)
positive	unrestricted	340 (45.5%)	354 (49.03%)	526 (58%)	455 (52.3%)	49 (77%)	102 (19%)

This indicates that although the choice of model specification has a large influence on the final number of distinct tokens selected by all the different specifications, there are core tokens that occur in all dictionaries. To gain some perspective on the bi-gram tokens that were selected for the dictionaries, we visually represent key words.

Using network graphs, figures (1) to (3) show the network among a bi-gram text sample for each of the outcome variables.

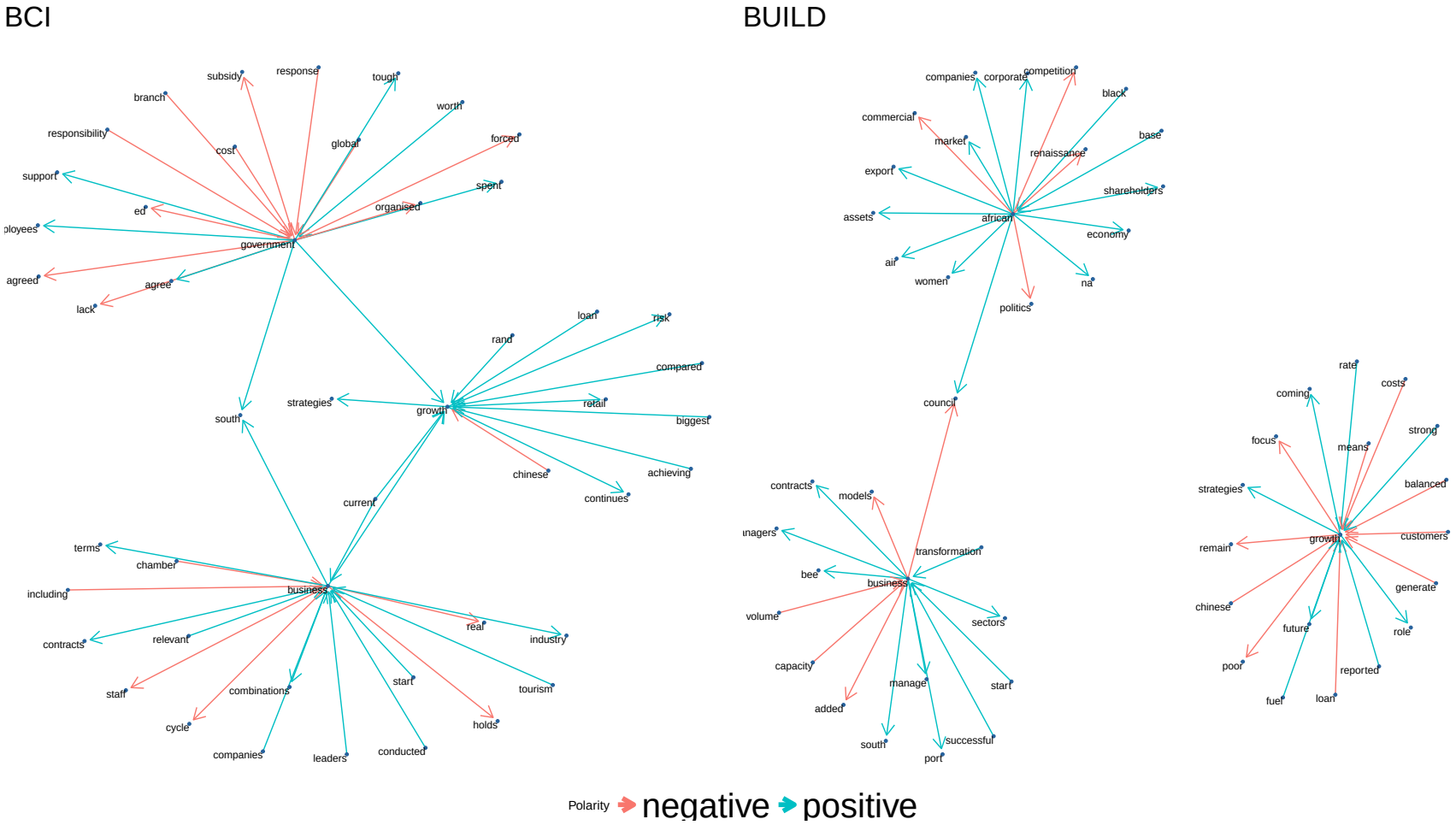


Figure 1: Network graph based off of bi-gram tokens filtered on the top three highest measures of betweenness centrality (BCI/Build).

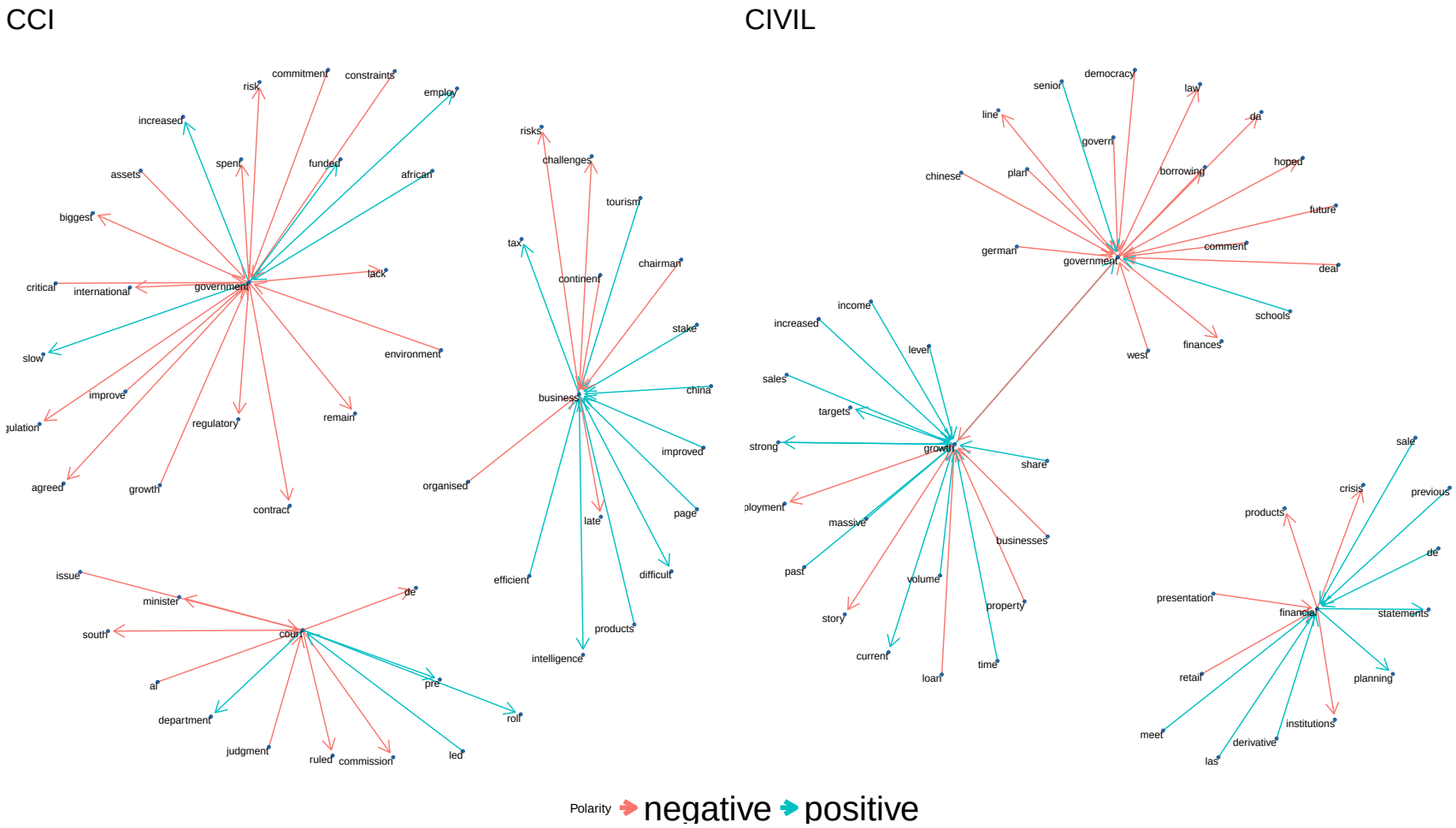


Figure 2: Network graph based off of bi-gram tokens filtered on the top three highest measures of betweenness centrality (CCI/Civil).

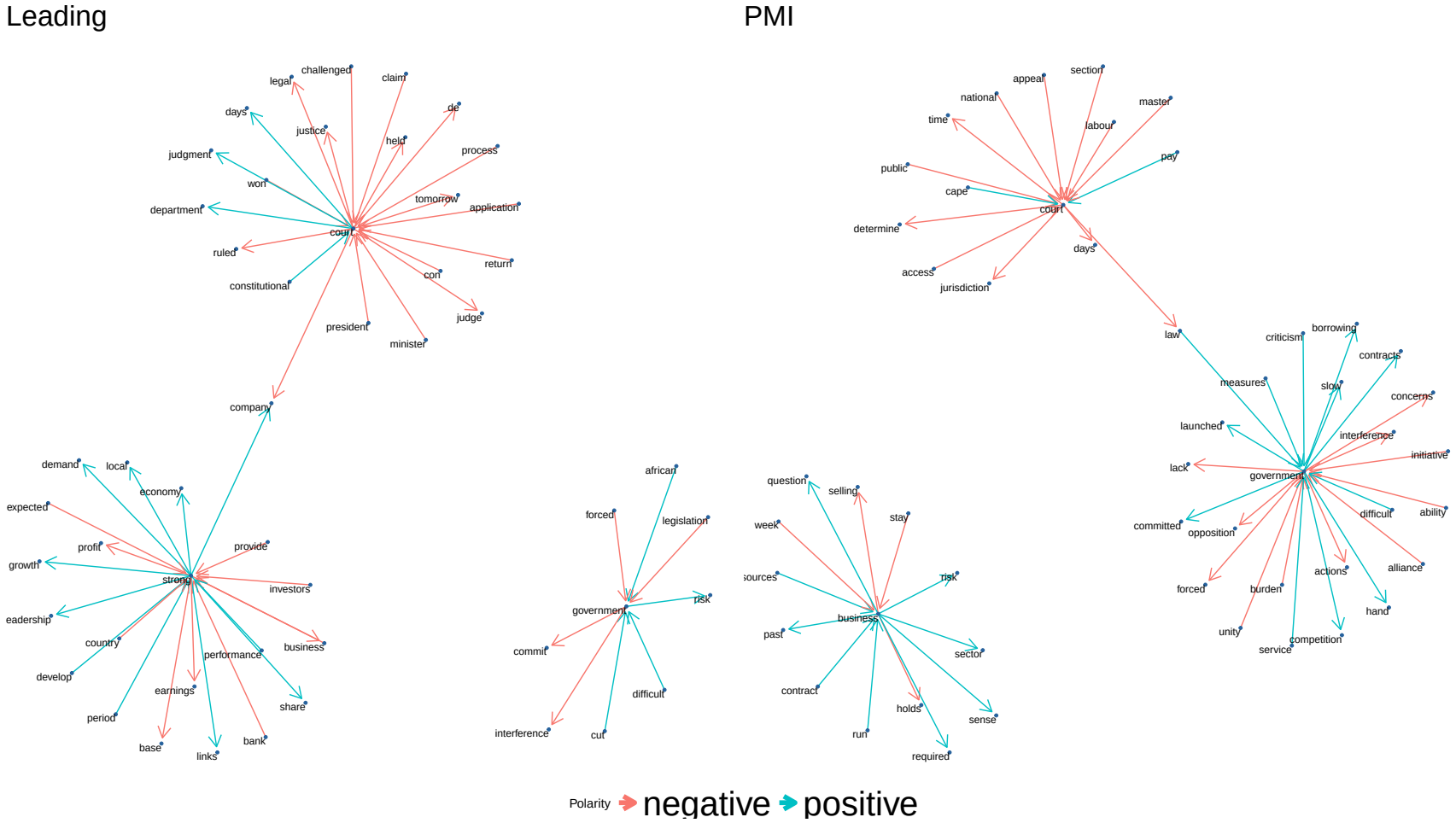


Figure 3: Network graph based off of bi-gram tokens filtered on the top three highest measures of betweenness centrality (Leading/PMI).

To construct the network graph, all dictionaries (for a given outcome) were collapsed and the three bi-grams with the highest degree of *betweenness* were kept.[10] The betweenness centrality measure indicates which tokens have a high flow of information passing through them and is principally based on shortest path algorithms. Thus, the tokens with a high degree of betweenness form central concepts key to the various dictionaries.

Figures (1) to (3) show that for government, the tokens tend to have a negative polarity, while growth has more positive tokens connected to it. This could be due to the bias found within the media that tends to report negative news relating to government activities more eagerly than positive news. Table 3 gives the rank of betweenness for the top ten tokens as per each of the different outcomes. The table contains 20 unique tokens that were isolated. Tokens that were selected across all the outcomes include 'business,' 'company,' 'government,' and 'market.' Although the token 'growth' does not appear in the PMI-generated dictionary, it is considered to be important for all other measures.

[10] See Freeman (1977) for the seminal work on betweenness centrality.

Table 3: Rank of the top ten words as per the betweenness centrality measure for each of the outcomes. In total, 20 unique tokens were isolated and of these 'business', 'company', 'government', and 'market' occurred in all outcome dictionaries.

token	BCI	BUILD	CCI	CIVIL	Leading	PMI
african	5	3	6	-	-	-
business	2	1	3	4	7	3
company	7	6	9	6	5	5
court	-	-	1	-	1	1
economic	-	10	-	7	-	-
financial	10	7	-	2	-	10
global	8	-	-	-	-	-
government	3	4	2	3	2	2
growth	1	2	4	1	8	-
investment	-	-	-	10	-	-
law	-	-	10	-	6	-
market	4	5	7	5	9	8
national	9	-	-	8	-	-
people	-	-	-	-	-	6
poor	-	-	-	-	10	-
risk	-	-	5	-	4	4
share	-	-	-	9	-	-
south	6	9	-	-	-	-
strong	-	8	8	-	3	9
time	-	-	-	-	-	7

We can see that for the BCI, CIVIL, and BUILD indices, the term growth had the highest (or second) highest measure of centrality, while for the CCI, SARB leading indicator, and PMI, the token 'court' had the highest value.

Examining figures (1) to (3) and table 3 confirms that the algorithm's variable selection method is isolating words that have meaning within the broader context of economic sentiment. Although these tokens or their polarity do not necessarily equate to their linguistic orientation, they are capturing some form of sentiment towards the economy, business, government, growth, and the markets. To evaluate how well these automated dictionaries capture sentiment, we constructed sentiment indices and analysed out-of-sample fits between the various confidence measures and the

constructed sentiment index.

5.1.1. Constructing and evaluating sentiment indices

This section deals with the construction and evaluation of various sentiment indices using the generated dictionaries. These indices are compared to a sentiment index that is constructed using the well-known Loughran & McDonald dictionary as a baseline. To evaluate these indices, we use the root mean squared error between the constructed indices and the respective confidence measure as an indication of fit:

$$RMSE = \sqrt{\frac{\sum_{t=1}^{T}(s_t - y_t)^2}{T}} \tag{3}$$

Where s_t is the sentiment index at t and y_t confidence measure.

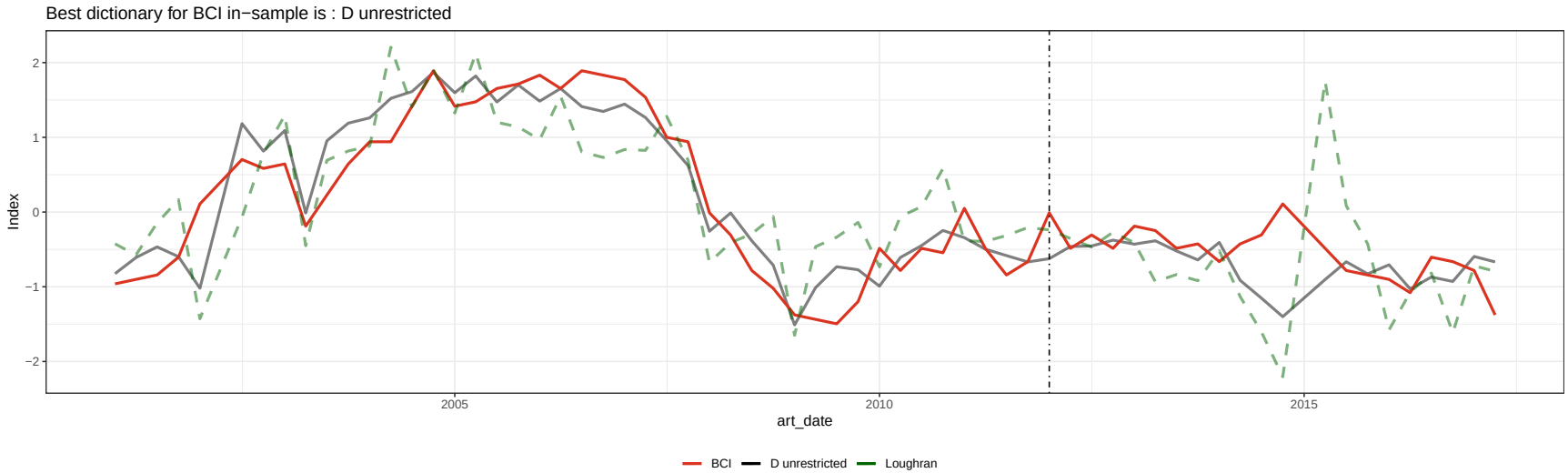
Following Odendaal et al. (2020), we constructed the indices using a net score. We identified the positive and negative words for all article $Nt \quad \forall \quad T$ using the generated word lists (dictionary). Conducting a simple word count that consists of the positive plus negative words, a sentiment index was created. We normalised the count so that it reflects the relative proportion of positive and negative words within a period:

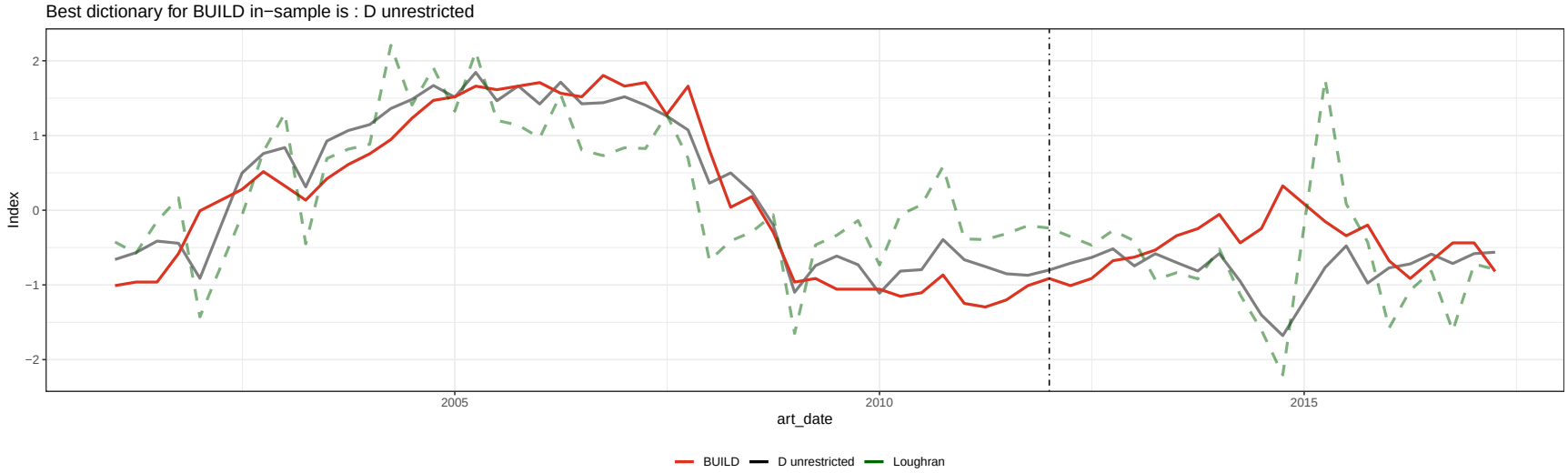
$$Pos_t = \frac{PositiveWords}{PositiveWords + NegativeWords} \quad Neg_t = \frac{NegativeWords}{PositiveWords + NegativeWords} \tag{4}$$

The overall sentiment index for a given time period t can then be defined as:

$$S_t = Pos_t - Neg_t \tag{5}$$

The resulting index is the net balance of positive and negative words within a quarter. Figure (4) shows the constructed indices, the sentiment index constructed using the baseline dictionary as well as the confidence measure.


Best dictionary for BCI in−sample is : D unrestricted
Index
art_date
2005
2010
2015
BCI
D unrestricted
Loughran


Best dictionary for BUILD in−sample is : D unrestricted
Index
art_date
2005
2010
2015
BUILD
D unrestricted
Loughran


Best dictionary for CCI in−sample is : E loughran
Index
2
1
0
−1
−2
2005
2010
2015
art_date
CCI
E loughran
Loughran


Best dictionary for CIVIL in−sample is : D unrestricted
Index
2
1
0
−1
−2
2005
2010
2015
art_date
CIVIL
D unrestricted
Loughran

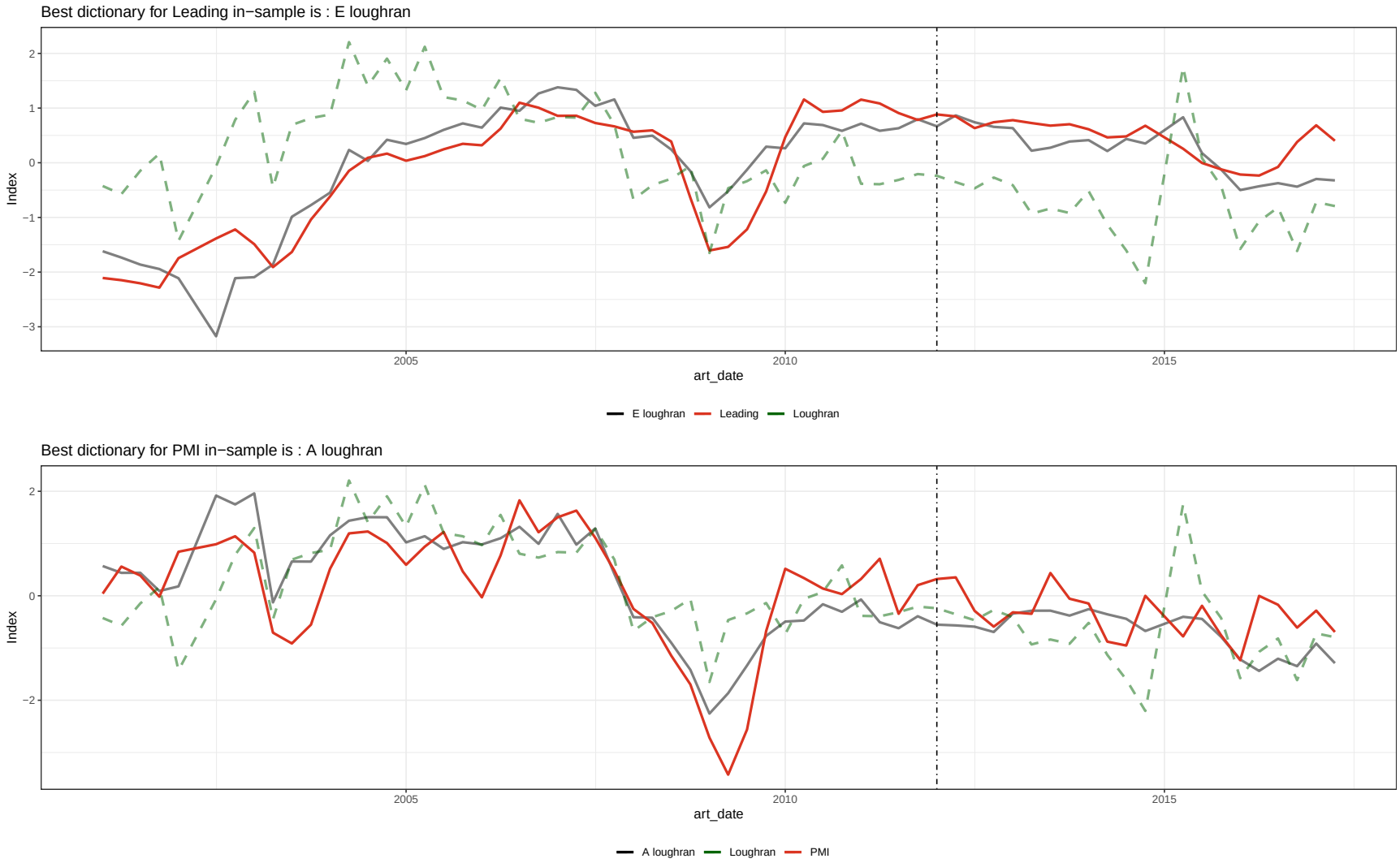


Figure 4: Figure illustrating the fit of the constructed sentiment indices with their respective confidence outcome measure.

For each of the series, the dictionary with the lowest RMSE in-sample is highlighted. For three of the indices, BCI, BUILD, and CIVIL, the best RMSE was achieved using specification D. This entailed including the lag (and change) of the underlying series in the predictor set, building 1000 trees, not using Loughran as a prior, and finally, setting the $mtry$ tuning parameter to $\sqrt{p}$. The second most used specification was E (CCI and SARB Leading) that uses the Loughran dictionary as a prior, does not include properties of the outcome series into the predictor set, only builds 500 trees, and uses a low number of variables to split on ($mtry = \sqrt{p}/3$). For the PMI, the sentiment index that had the lowest in-sample RMSE was A, the Loughran prior specification. This specification is identical to the specification that had the lowest RMSE for the CCI and the SARB leading indicator, but it had the $mtry$ parameter as high $= \sqrt{p}$.

To evaluate the underlying relationship between the generated series and the confidence measure, we calculated the cross-correlation between the series out-of-sample. Cross-correlation can be defined as a measure of the similarity between two series as a function of the lag of the predictor relative to the outcome. We conducted this test in order to assess whether generated sentiment indices perhaps lag or lead the confidence measure. We used the specification which had the lowest RMSE in-sample and the traditional Loughran McDonald dictionary and calculated the cross-correlation between the series and the outcome. The series were all tested for stationarity before estimating the cross-correlation and found to be non-stationary, so all series were made stationary using first log difference. The series were tested using a maximum lag of up to four quarters. Figure (5) shows how the sentiment indices correlate with changes observed in the CCI, leading indicator, and PMI.

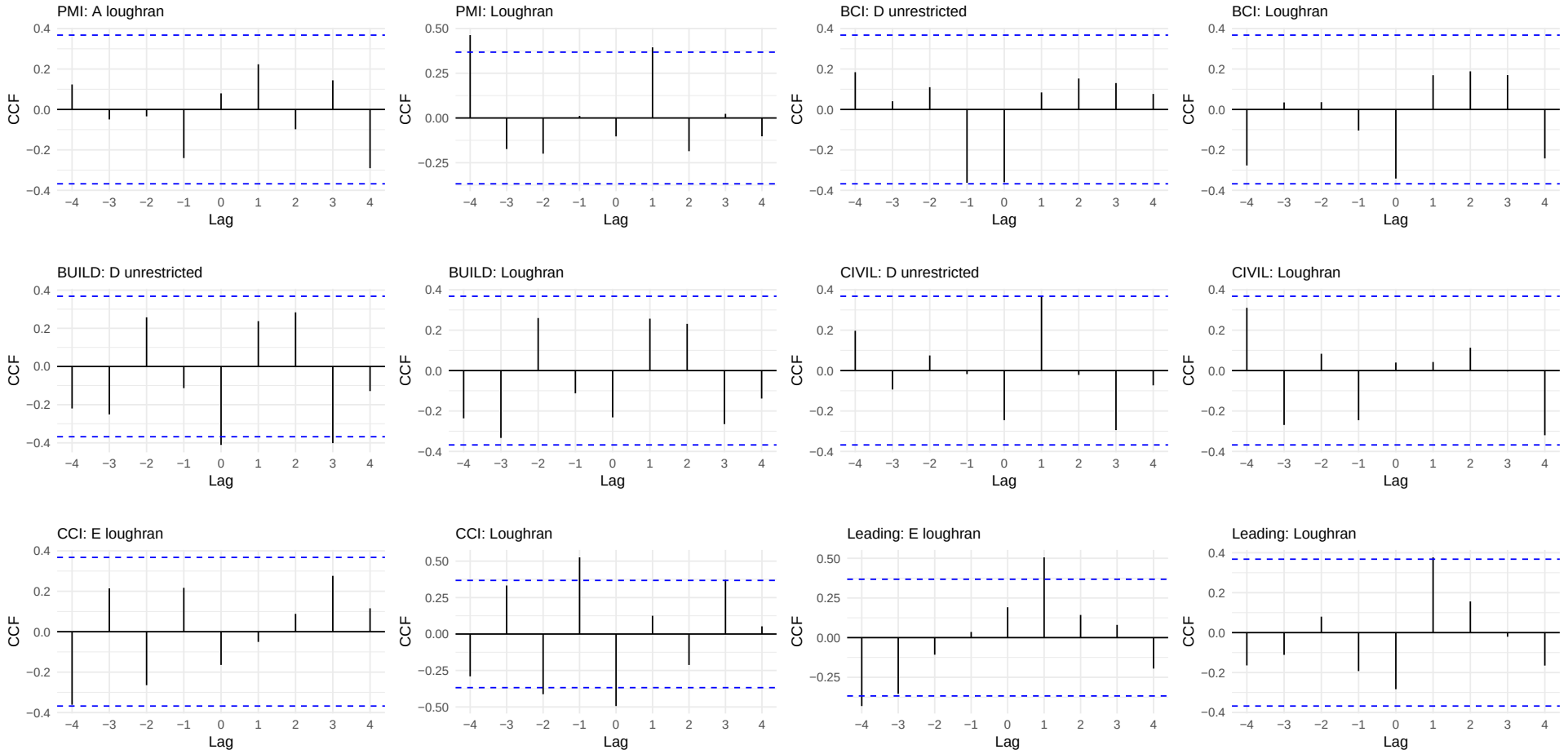


Figure 5: Cross-correlation between the constructed sentiment indices with their respective confidence outcome measure. The statistical significance of the correlation between the series is tested at the 10% level and is indicated by the dotted lines.

In the case of the CCI, the correlation is negative, while the correlation is positive for the leading indicator and the PMI. For the generated dictionaries, fewer of the correlations were statistically significant. A contemporaneous negative correlation is seen between the BUILD confidence index and specification *D unrestricted*. The other significant relationship is between the leading indicator and *E unrestricted*. Although correlation is a way to understand the dynamic relationship between two series, it does not tell us how well the sentiment indices fit the original confidence indicators. To evaluate this, we looked at the out-of-sample fits.

Given the nature of the algorithm and the possibility that it is overfitting the series in-sample, we could not compare the sentiment indices generated from the newly generated dictionaries with the well-known Loughran dictionary. We turned to an out-of-sample RSME evaluation of the various series.[11] Table (4) contains the RMSE measures as well as the percentage difference of each specification to the specification that achieved the lowest RMSE in-sample. Using BCI as an example, the specification that achieved the lowest RMSE in-sample was *D unrestricted*. This same specification achieved an RMSE of 0.475 out-of-sample. Using this number as the base, we compare the RMSE of the other specifications with 0.475 to evaluate the best in-sample specification's performance out-of-sample. On average, the domain-specific dictionaries generated for the BCI only differed by -2% out-of-sample, while for the SARB leading indicator, the difference was much larger at 37.6%.

[11] The author also calculated various other error measures which are available on request: ME, MAE, MPE, and MAPE. The findings remained irrespective of error measurement.

Table 4: The RMSE measure as calculated by comparing the generated series and the actual indices based on 20 quarterly out-of-sample observations from 2012 to 2017. The values represent the RMSE measures per specification as well the percentage difference in error of each specification with the specification that had the lowest RMSE in-sample (in brackets).

Specification	Type	Inc lag	Tree size	Mtry size	Bci	Build	Cci
A	loughran	FALSE	500	high	0.509 (7.2%)	0.493 (-19.0%)	0.693 (4.0%)
A	unrestricted	FALSE	500	high	0.430 (-9.6%)	0.581 (-4.6%)	0.893 (34.0%)
B	loughran	TRUE	500	high	0.482 (1.4%)	0.437 (-28.2%)	0.683 (2.4%)
B	unrestricted	TRUE	500	high	0.500 (5.2%)	0.627 (3.0%)	0.840 (25.8%)
C	loughran	FALSE	1000	high	0.445 (-6.4%)	0.483 (-20.8%)	0.662 (-0.8%)
C	unrestricted	FALSE	1000	high	0.500 (5.2%)	0.627 (3.0%)	0.883 (32.4%)
D	loughran	TRUE	1000	high	0.454 (-4.4%)	0.455 (-25.2%)	0.690 (3.4%)
D	unrestricted	TRUE	1000	high	0.475 (-)	0.609 (-)	0.735 (10.2%)
E	loughran	FALSE	500	low	0.471 (-0.8%)	0.454 (-25.4%)	0.667 (-)
E	unrestricted	FALSE	500	low	0.437 (-8.0%)	0.644 (5.6%)	0.884 (32.4%)
F	loughran	TRUE	500	low	0.428 (-10.0%)	0.446 (-26.8%)	0.676 (1.2%)
F	unrestricted	TRUE	500	low	0.452 (-4.8%)	0.637 (4.6%)	0.942 (41.2%)
G	loughran	FALSE	1000	low	0.435 (-8.4%)	0.436 (-28.4%)	0.711 (6.6%)
G	unrestricted	FALSE	1000	low	0.482 (1.4%)	0.603 (-1.0%)	0.872 (30.8%)
H	loughran	TRUE	1000	low	0.495 (4.2%)	0.444 (-27.0%)	0.675 (1.2%)
H	unrestricted	TRUE	1000	low	0.451 (-5.0%)	0.610 (0.2%)	0.788 (18.2%)
loughran	-	-	-	-	0.880 (85.2%)	0.907 (49.0%)	1.019 (52.8%)
Mean diff	-	-	-	-	-2.0%	-11.8%	15.2%

Specification	Type	Inc lag	Tree size	Mtry size	Civil	Leading	Pmi
A	loughran	FALSE	500	high	0.586 (-23.2%)	0.654 (59.2%)	0.616 (-)
A	unrestricted	FALSE	500	high	0.758 (-0.6%)	0.416 (1.2%)	0.553 (-10.4%)
B	loughran	TRUE	500	high	0.627 (-17.6%)	0.756 (84.0%)	0.606 (-1.6%)
B	unrestricted	TRUE	500	high	0.701 (-8.0%)	0.394 (-4.0%)	0.502 (-18.6%)
C	loughran	FALSE	1000	high	0.583 (-23.6%)	0.737 (79.4%)	0.623 (1.2%)
C	unrestricted	FALSE	1000	high	0.764 (0.2%)	0.394 (-4.2%)	0.557 (-9.6%)
D	loughran	TRUE	1000	high	0.591 (-22.4%)	0.897 (118.2%)	0.660 (7.0%)
D	unrestricted	TRUE	1000	high	0.762 (-)	0.510 (24.0%)	0.525 (-14.8%)
E	loughran	FALSE	500	low	0.586 (-23.0%)	0.411 (-)	0.660 (7.0%)
E	unrestricted	FALSE	500	low	0.768 (0.8%)	0.416 (1.2%)	0.555 (-10.0%)
F	loughran	TRUE	500	low	0.575 (-24.6%)	0.771 (87.6%)	0.645 (4.6%)
F	unrestricted	TRUE	500	low	0.863 (13.2%)	0.439 (6.8%)	0.569 (-7.6%)
G	loughran	FALSE	1000	low	0.574 (-24.6%)	0.647 (57.6%)	0.640 (3.8%)
G	unrestricted	FALSE	1000	low	0.752 (-1.4%)	0.360 (-12.4%)	0.569 (-7.6%)
H	loughran	TRUE	1000	low	0.569 (-25.4%)	0.843 (105.2%)	0.623 (1.0%)
H	unrestricted	TRUE	1000	low	0.757 (-0.8%)	0.404 (-1.6%)	0.565 (-8.4%)
loughran	-	-	-	-	1.008 (32.2%)	1.437 (249.6%)	0.942 (52.8%)
Mean diff	-	-	-	-	-11.4%	37.6%	-4.0%

In comparison, the differences observed when comparing the Loughran dictionary's fit are much larger. The average difference among all the outcomes is 85%. This would indicate that the domain-specific dictionaries significantly improved on the out-of-sample fit when compared to the financial dictionary. The largest difference observed in RMSE measures was when comparing the fit of the SARB leading indicator. Here, the domain-specific dictionary decreased the RMSE by a factor 2.5 out-of-sample.

This led us to ask whether the specifications themselves are different and whether the choice of specification makes a significant difference to the out-of-sample performance of the sentiment index. To answer this question, we employed a non-parametric statistical method in order to test the hypothesis that the mean RMSE values of the difference specifications are the same. We used a pairwise Wilcoxon ranks sum test and compared means between group levels. The null hypothesis for the test is: true location shift is not greater than zero (i.e., the RMSE values are roughly the same).

Table (5) shows the p-values obtained from all the pairwise tests conducted. The insignificance of all of the values indicates that there is no statistically significant difference in the RMSE measures across the different specifications of the models. This finding highlights the fact that on average, the specification of the model does not have a significant impact on the mean RMSE measures produced out-of-sample. It has to be clarified that although this was the case for the outcomes presented in this paper, the finding is unlikely to generalise across different datasets. Further research is needed to understand the effect of each subjective choice in the final generated dictionary.

Table 5: Results from pairwise Wilcoxon ranks sum test. The high p-values among all the tests indicate that there is no statistically significant difference in the RMSE measures across the different specifications.

Variables	Inc lag false	Inc lag true	Mtry size high	Mtry size low	Tree size 1000
Inc lag: TRUE	0.283	-	-	-	-
Mtry size: high	0.316	0.525	-	-	-
Mtry size: low	0.464	0.7	0.665	-	-
Tree size: 1000	0.35	0.56	0.536	0.372	-
Tree size: 500	0.426	0.669	0.633	0.464	0.597

Apart from testing the specifications among themselves, the Wilcoxon test was also employed to test whether a statistically significant difference exists between the unrestricted, Loughran prior, and the traditional Loughran sentiment indices. Figure (6) shows the boxplot of the pairwise Wilcoxon

test between the different model types as well as the global test. The unrestricted and Loughran prior model types show no statistical difference between their mean RMSEs. There is however a statistically significant difference between the Loughran dictionary and the model generated dictionaries. This is confirmed by the small p-value of the pairwise (p = 1.5e-07, p = 2.3e-06) and global test (p = 3.4e-04).

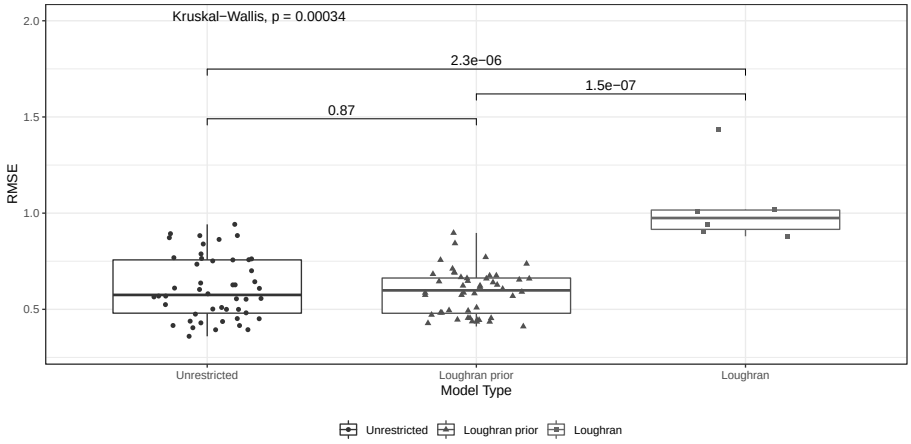


Figure 6: Pairwise Wilcoxon test between the different model types.

However, within the machine generated dictionaries, the Loughran prior has a smaller deviation of RMSE measures, $\sigma = 0.11$, than its unrestricted counterpart, $\sigma = 0.16$. These results indicate that although the mean errors from the unrestricted and Loughran prior are not statistically different from one another, generated dictionaries that use a domain-specific dictionary as a prior could deliver the same results with much less noise. Using a Fligner-Killeen test, the null hypothesis of equal variance was rejected at the 5% level with a p-value of 0.0168. These findings motivate the use of a domain-specific dictionary, that has been proven effective through human verification, as a starting point when using machine learning algorithms to generate dictionaries for sentiment analysis.

6. Conclusion

In this paper, machine learning was presented as one way to create a domain-specific dictionary. This approach has been shown to be able to create sentiment lexicons tailored to a specific need.

These dictionaries, being less subjective, are more easily tested and replicable. Specific word lists also contribute to transparency since it is easy for other researchers to replicate the results.

Tokens (words) were statistically selected using recursive feature elimination to generate a domain-specific dictionary from a corpus of text. The machine-generated dictionaries consist of both uni-grams and bi-grams. The inclusion of bi-grams add context to the sentiment dictionary, ensuring that the tokens themselves are less ambiguous. To try and understand the bi-grams in the dictionaries, we visually show that they contain central ideas (topics) that help in capturing sentiment for each of the confidence measures. These ideas ranged from issues in government, business, and growth, as well as what is happening in the markets. This concentration of tokens around key concepts substantiates the findings of Sims (2003) and the hypothesis around the rational inattention model. Having to focus on the economy and its intricacies has an opportunity cost connected to it; in turn, the agents would rather process information on economic activity through concentrated sources, such as the media (and expert reporting), where key information on business, markets, government, etc. can be more easily digested. This distillation of key information can be tracked in a generalised construct known as sentiment indices.

The results of this paper show that the indices constructed from machine-generated dictionaries have a better fit with the multiple indicators investigated compared to the sentiment index constructed from a commonly used financial dictionary. These domain-specific sentiment indices also show a significantly lower root mean squared error (RMSE) in a five-year holdout sample period from 2012 to 2017. The largest improvement was observed for the leading indicator, where the domain-specific dictionary improved the fit by a factor of 2.5.

These results support the case for domain-specific dictionaries being able to pick up nuances found within domain-specific topic news. The results, however, suggest that having a manually generated dictionary act as a prior narrows the tokens[12] that the Random Forest has to search over, while maintaining the same lower RMSE out-of-sample as an unrestricted model with all tokens. Employing a manually generated dictionary such as that of Loughran and McDonald (2011) also decreases the computational burden on the pre-processing and estimation of the dictionaries. Another finding of the paper relates to practical considerations when implementing the Random Forest algorithm tuning parameters. In the case of this paper's data, the tuning parameters had

[12] See section 4 for reference.

no statistical difference in the RMSE across all different specifications. Although this was the finding for this paper's specific research design, it is a finding that needs further research to validate and understand. A better understanding of the dynamics between the tuning parameters and the resulting dictionary could result in much more robust dictionary generation, especially in an operational setting.

References

Akerlof, G.A., Dickens, W.T., Perry, G.L., Bewley, T.F., Blinder, A.S., 2000. Near-rational wage and price setting and the long-run phillips curve. Brookings papers on Economic Activity 2000, 1–60.

Athey, S., Tibshirani, J., Wager, S., 2016. Generalized random forests.

Biau, O., D'Elia, A., 2009. Euro area GDP forecasting using large survey datasets, in: A Random Forest Approach.

Breiman, L., 2001. Random forests. Machine Learning 45, 5–32.

Carroll, C.D., 2003. Macroeconomic expectations of households and professional forecasters. the Quarterly Journal of Economics 118, 269–298.

Freeman, L.C., 1977. A set of measures of centrality based on betweenness. Sociometry 40, 35–41.

Friedman, J., Hastie, T., Tibshirani, R., 2001. The elements of statistical learning. Springer series in statistics New York.

Genuer, R., Poggi, J.-M., Tuleau-Malot, C., 2010. Variable selection using random forests. Pattern Recognition Letters 31, 2225–2236.

Goidel, R.K., Langley, R.E., 1995. Media coverage of the economy and aggregate economic evaluations: Uncovering evidence of indirect media effects. Political Research Quarterly 48, 313–328.

Grömping, U., 2009. Variable importance assessment in regression: Linear regression versus random forest. The American Statistician 63, 308–319.

Guyon, I., Elisseeff, A., 2003. An introduction to variable and feature selection. Journal of Machine Learning Research 3, 1157–1182.

Henry, E., 2008. Are investors influenced by how earnings press releases are written? The Journal of Business Communication (1973) 45, 363–407.

Kalamara, E., Turrell, A., Redl, C., Kapetanios, G., Kapadia, S., 2020. Making text count: Economic forecasting using newspaper text. Bank of England.

Kane, M.J., Price, N., Scotch, M., Rabinowitz, P., 2014. Comparison of ARIMA and random forest time series models for prediction of avian influenza H5N1 outbreaks. BMC Bioinformatics 15, 276.

Kuhn, M., Johnson, K., 2013. Applied predictive modeling. Springer.

Labille, K., Gauch, S., Alfarhood, S., 2017. Creating domain-specific sentiment lexicons via text mining, in: Proc. Workshop Issues Sentiment Discovery Opinion Mining (WISDOM).

Lamla, M.J., Lein, S.M., 2014. The role of media for consumers' inflation expectation formation. Journal of Economic Behavior & Organization 106, 62–77.

Leduc, S., Sill, K., Stark, T., 2007. Self-fulfilling expectations and the inflation of the 1970s: Evidence from the livingston survey. Journal of Monetary Economics 54, 433–459.

Liu, B., 2012. Sentiment analysis and opinion mining. Synthesis lectures on human language technologies 5, 1–167.

Loughran, T., McDonald, B., 2011. When is a liability not a liability? Textual analysis, dictionaries, and 10-ks. The Journal of Finance 66, 35–65.

Mankiw, N.G., Reis, R., 2002. Sticky information versus sticky prices: A proposal to replace the new keynesian phillips curve. The Quarterly Journal of Economics 117, 1295–1328.

Meinshausen, N., 2006. Quantile regression forests. Journal of Machine Learning Research 7, 983–999.

Nadeau, R., Niemi, R.G., Fan, D.P., Amato, T., 1999. Elite economic forecasts, economic news, mass economic judgments, and presidential approval. The Journal of Politics 61, 109–135.

Nielsen, F.Å., 2011. Afinn. Richard Petersens Plads, Building 321.

Odendaal, H., Reid, M., Kirsten, J.F., 2020. Media-based sentiment indices as an alternative measure of consumer confidence. South African Journal of Economics 88, 409–434.

Pröllochs, N., Feuerriegel, S., Neumann, D., 2015. Generating domain-specific dictionaries using bayesian learning. ECIS Completed Research Papers 144.

Sims, C.A., 2003. Implications of rational inattention. Journal of monetary Economics 50, 665–690.

Soroka, S.N., Stecula, D.A., Wlezien, C., 2015. It's (change in) the (future) economy, stupid: Economic indicators, the media, and public opinion. American Journal of Political Science 59, 457–474.

Strobl, C., Boulesteix, A.-L., Zeileis, A., Hothorn, T., 2007. Bias in random forest variable importance measures: Illustrations, sources and a solution. BMC bioinformatics 8, 25.

Takamura, H., Inui, T., Okumura, M., 2005. Extracting semantic orientations of words using spin model, in: Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics. Association for Computational Linguistics, pp. 133–140.

Tyralis, H., Papacharalampous, G., 2017. Variable selection in time series forecasting using random forests. Algorithms 10, 114.

Wright, M.N., Ziegler, A., 2015. Ranger: A fast implementation of random forests for high dimensional data in c++ and r. arXiv preprint arXiv:1508.04409.