



N-Gram Feature for Comparison of Machine Learning Methods on Sentiment in Financial News Headlines

Arif Mudi Priyatno¹, Fahmi Iqbal Firmananda²

^{1,2}Bisnis Digital, Fakultas Ekonomi dan Bisnis, Universitas Pahlawan Tuanku Tambusai
¹arifmudi@universitaspahlawan.ac.id, ²fahmiiqbalfirmananda@universitaspahlawan.ac.id

Abstract

Sentiment analysis is currently widely used in natural language processing or information retrieval applications. Sentiment analysis analysis can provide information related to outstanding financial news headlines and provide input to the company. Positive sentiment will also have a good impact on the development of the company, but negative sentiment will damage the company's reputation. This will affect the company's development. This study compares machine learning methods on financial news headlines with n-gram feature extraction. The purpose of this study was to obtain the best method for classifying the headline sentiment of the company's financial news. The machine learning methods compared are Multinomial Naïve Bayes, Logistic Regression, Support Vector Machine, multi-layer perceptron (MLP), Stochastic Gradient Descent, and Decision Trees. The results show that the best method is logistic regression with a percentage of f1-measure, precision, and recall of 73.94 %, 73.94 %, and 74.63 %. This shows that the n-gram and machine learning features have successfully carried out sentiment analysis.

Keywords: N-Gram, Multinomial Naïve Bayes, Logistic Regression, Support Vector Machine, multi-layer perceptron, Stochastic Gradient Descent, Decision Trees, sentiment analyst

1. Introduction

Information and communication technology are currently developing very rapidly. Various things can be done with technology, especially with the presence of artificial intelligence. One example that can be done is conducting a sentiment analysis on news headlines. Sentiment analysis on text can be done using text mining.

Sentiment analysis is the process of categorizing opinions or topics in the text. This process is to determine whether opinions or topics are positive, negative, or neutral [1]. Sentiment analysis research has been carried out in various sectors, such as finance [2], politics [3], companies [4], and so on. Sentiment analysis on financial and company news will have an effect on the credibility of a company. Sentiment has a big influence on companies that are open or listed on the stock exchange.

This study compares machine learning methods for sentiment analysis with the n-gram feature in the case of financial news and corporate headlines. Data from research [5] with a total of 4846 titles was used. The number of neutral sentiments is 2879, the number of positive sentiments is 1363, and the number of negative

sentiments is 604. The data is preprocessed by deleting things that are not needed. Preprocessing results are given weight by the n-gram method. The n-gram features are analyzed by sentiment analysis using several machine learning methods, including Multinomial Nave Bayes, Logistic Regression, Support Vector Machine, multi-layer perceptron (MLP), Stochastic Gradient Descent, and Decision Trees. Sentiment analysis performance is calculated using precision, recall, and f1-measure.

Related research is [6] conducting sentiment analysis on news articles based on lexicon. The data used is BBC News from 2004 to 2005 (<http://mlg.ucd.ie/datasets/bbc.html>). The study concluded that articles on business and sports topics had positive sentiments, while topics related to entertainment and sports had negative sentiments.

Research [7] investigated the metrics of measuring sentiment analysis in news articles. The weighting method used is term frequency-inverse document frequency (TF-IDF). The classification method uses Gaussian Naive Bayes classifiers and linear support vector machines (linear SVM). The best results were obtained using the linear SVM method, with 62 percent, while Gaussian Bayes got 61 percent.

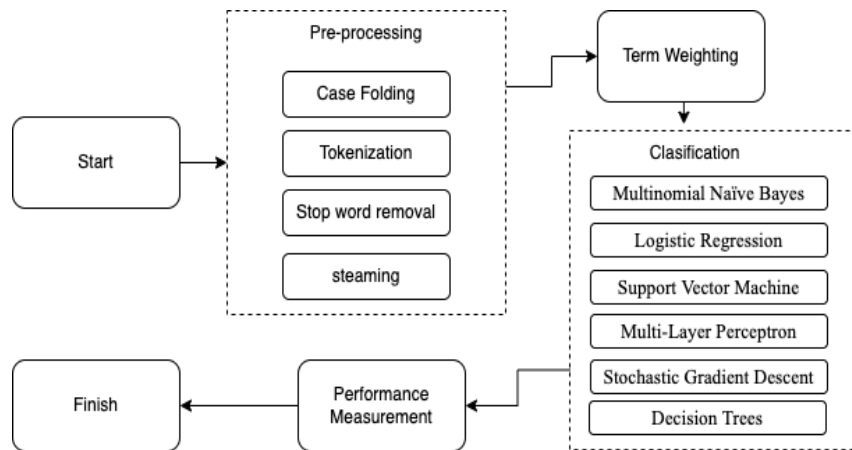


Figure 1. Research Stages

Research [8] conducted a sentiment analysis on financial news using VADER (Valence Aware Dictionary and Sentiment Reasoner). This study aims to show the effect of financial news sentiment on changes in stock market prices.

Research [9] conducted a comparison of analytical methods on news headlines in the Netherlands. The comparisons are between manual annotation, crowd coding, numerous dictionaries, machine learning, and deep learning. The conclusion is that the best results are done by humans or crowd coding. Numerous dictionaries indicate that the level of validity is acceptable, and machine learning and deep learning are substantially better than numerous dictionaries but are still far from human performance.

2. Research Methods

The research process includes data collection, preprocessing, weighting using n-grams, sentiment classification, and performance measurement. Figure 1 shows the stages of the research carried out.

2.1. Data

The data is the text of financial news and company press releases. Sentiment marking was carried out by a group of 16 annotators with adequate business education backgrounds. The data uses research [5] with a total of 4846 news titles. The number of neutral sentiments is 2879, the number of positive sentiments is 1363, and the number of negative sentiments is 604.

2.2. Pre-processing

This stage is carried out to prepare data that can be used in the next stages. Preprocessing cleans text data from common words (tokens) that do not have the necessary meaning. This is done to reduce noise in text data. Figure 1 shows the processes contained in the preprocessing, namely case folding, tokenization, stopword removal, and steaming. Case folding is the

first step to making letters uniform by changing all uppercase letters to lowercase letters. For example, "According to Gran, the company has no plans to move all production to Russia, although that is where the company is growing" becomes "according to gran, the company has no plans to move all production to Russia, although that is where the the company is growing". Tokenization is the stage of cutting sentences into smaller tokens (terms). For example "according to gran, the company has no plans to move all production to russia, although that is where the company is growing." be ["according", "to", "gran", "the", "company", "has", "no", "plans", "to", "move", "all", "production", "to", "russia", "although", "that", "is", "where", "company", "is", "growing"]. Stop word removal is the stage of removing general terms and has no effect. Examples of deleted terms are 'its', 'again', 'for', 'myself', 'his', and so on [10]. Steaming is the process of returning terms to their basic words [11]. For example, "plans" becomes "plan", "growing" becomes "grow".

2.3. Term Weighting

An N-gram is a cut-up sentence or string into smaller ones according to the specified N characters. Gram is defined as a sub sequence of N characters that are worked out [12]. The n-gram method is used with the aim of giving the meaning of a sequence of words or characters in a sentence. The n-gram method is used to take n character pieces from sentences that are continuous. The continuity in question is from the beginning to the end of the document.

Tabel 1. Contoh n-gram

N-gram	Result
Unigram	: 'a', 'c', 'c', 'o', 'r', 'd', 'g', 'r', 'a', 'n'
Bigram	: 'a', 'ac', 'cc', 'co', 'or', 'rd', 'dg', 'gr', 'ra', 'an', 'n_'
Trigram	: '_ac', 'acc', 'cco', 'cor', 'ord', 'rdg', 'dgr', 'gra', 'ran', 'an_', 'n_'
Quadgram	: '_acc', 'acco', 'ccor', 'cord', 'ordg', 'rdgr', 'dgra', 'gran', 'ran_', 'an_', 'n_'

Table 2. N-Gram on Research

N-gram	Result
Unigram	: 'according', 'gran', 'company', 'plan', 'move', 'production', 'russia', 'although', 'company', 'grow'
Bigram	: 'according gran', 'gran company', 'company plan', 'plan move', 'move production', 'production russia', 'russia although', 'although company', 'company grow'
Trigram	: 'according gran company', 'gran company plan', 'company plan move', 'plan move production', 'move production russia', 'production russia although', 'russia although company', 'although company grow'
Quadgram	: 'according gran company plan', 'gran company plan move', 'company plan move production', 'plan move production russia', 'move production Russia although', 'production russia although company', 'russia although company grow'

An N-gram is classified based on n characters. In general, the n-gram is done by adding additional blanks at the beginning and at the end [13]. For example, the sentence "accord gran" is processed by n-grams, the blank is symbolized by "_", resulting in n-grams in Table 1.

N-Gram has the advantage that matching every word can be used even if there is an interpretation of certain codes, such as postal codes. In natural language processing applications, certain numbers and codes are considered "noise". This is considered an advantage because of the decomposition of small parts that still have meaning. This textually has no effect.

In this study, the n-gram used is not broken down into small characters, but the n-gram is used to break sentences into words. An example of the sentence "according to the grand company plan to move production in Russia although the company grows" is carried out by the n-gram process and the results can be seen in Table 2. The term weighting in this study uses the number of grams that appear, or better known as term frequency (TF).

2.4. Sentiment Classification

This study compares several classification methods. The methods used are multinomial nave Bayes [14][15], logistic regression [16][17], support vector machine [18][19][20], multi-layer perceptron [21][22][23][24], stochastic gradient descent [25][26], and decision tree [27][28].

Table 3 Data Latih dan Data Uji

No	Persentase	Data Latih	Data Uji	Total
1	70 %	3392	1454	4868
2	80 %	3876	970	4868
3	90 %	4361	485	4868

2.5. Performance Measurement

Performance evaluation in this study uses a confusion matrix. Performance evaluations based on the confusion matrix include precision, recall, and f1-measure [11][29].

$$recall = \frac{TP}{TP+FN} \quad (1)$$

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

$$F1 - measure = \frac{2 \times precision \times recall}{precision+recall} \quad (3)$$

3. Results and Discussions

The data used in this study amounted to 4846 financial news headlines. The number of neutral sentiments is 2879, the number of positive sentiments is 1363, and the number of negative sentiments is 604. The data was divided into training data and test data with percentages of 70:30, 80:20, and 90:10. Table 3 shows the results of the distribution of training data and test data.

The data is pre-processed to clean text data. The pre-processing processes include case folding, tokenization, stop word removal, and steaming. Table 4 shows the results of the pre-processing.

The results of preprocessing are sentiment classification with several machine learning methods. Tables 5, 6, and 7 show the classification results.

The multinomial method gets the best results when the percentage division is 70:30, which is f1-measure 69.79 percent. The logistic regression method got the best results in the 80:20 division, namely an f1-measure of 73.94 percent. The support vector machine method gets the best results in the 80:20 division, which is 70.07 percent.

The multi-layer perceptron method got the best results at 90:10 division, which is 72.76 percent. This is because the multi-layer perceptron method requires a lot of training data in order to maximize its hidden layer to find out its classification. If the amount of training data is small, the hidden layer multi-layer perceptron will not run optimally. In the future, if you want to use multi-layer perceptrons, please increase the training data, either by adding manuals or by using data augmentation methods.

The stochastic gradient descent method gets the best results at a 90:10 division, which is 71.16 percent. The decision tree method got the best results in the 70:30 division, which was 67.99 percent. Overall, the best results were obtained by the logistic regression method. The difference in the results that are not far shows that the process method is appropriate and there is no underfitting or overfitting. This good result can be optimized again by using a better feature extraction method such as using fasttext weighting, glove or

Table 5 Result Percentage 70:30

Method	F-1 Measure	Precision	Recall
Multinomial NaiveBayes	69,79	69,61	70,56
Logistic Regression	73,93	73,27	73,93
Support Vector Machine	66,65	70,02	70,56
Multi-Layer perceptron	70,27	70,04	70,63
Stochastic Gradient Descent	68,89	68,65	69,25
Decision Trees	67,99	68,22	67,81

Table 6 Result Percentage 80:20

Method	F-1 Measure	Precision	Recall
Multinomial NaiveBayes	69,73	69,85	71,03
Logistic Regression	73,94	73,94	74,63
Support Vector Machine	70,07	75,29	73,60
Multi-Layer perceptron	69,32	69,28	69,38
Stochastic Gradient Descent	70,27	70,17	70,41
Decision Trees	67,27	67,03	67,62

Table 7 Result Percentage 90:10

Method	F-1 Measure	Precision	Recall
Multinomial NaiveBayes	68,65	69,27	69,69
Logistic Regression	73,70	73,81	74,43
Support Vector Machine	68,91	71,71	71,75
Multi-Layer perceptron	72,76	72,79	72,99
Stochastic Gradient Descent	71,16	71,08	71,75
Decision Trees	66,12	65,84	66,60

others. This is, of course, by adding the transfer learning that has been done.

4. Conclusion

This study compares machine learning methods for sentiment classification with n-gram feature extraction. The data is used for a total of 4868 news headlines. The n-gram weighting and classification with machine learning showed good results. The highest results were obtained, namely f1-measure, precision, and recall of 73.94, 73.94, and 74.63. Based on the results of the tests that have been carried out, n-gram is able to extract features from existing data.

Reference

- [1] S. Kurniawan, W. Gata, D. A. Puspitawati, N. , M. Tabrani, and K. Novel, "Perbandingan Metode Klasifikasi Analisis Sentimen Tokoh Politik Pada Komentar Media Berita Online," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 3, no. 2, pp. 176–183, 2019, doi: 10.29207/resti.v3i2.935.
- [2] K. Mishev, A. Gjorgjevikj, I. Vodenska, L. T. Chitkushev, and D. Trajanov, "Evaluation of Sentiment Analysis in Finance: From Lexicons to Transformers," *IEEE Access*, vol. 8, pp. 131662–131682, 2020, doi: 10.1109/ACCESS.2020.3009626.
- [3] E. Kušen and M. Strembeck, "Politics, sentiments, and misinformation: An analysis of the Twitter discussion on the 2016 Austrian Presidential Elections," *Online Soc. Networks Media*, vol. 5, pp. 37–50, 2018, doi: 10.1016/j.osnem.2017.12.002.
- [4] S. M. Shuhidan, S. R. Hamidi, S. Kazemian, S. M. Shuhidan, and M. A. Ismail, "Sentiment Analysis for Financial News Headlines using Machine Learning Algorithm," in *Advances in Intelligent Systems and Computing*, vol. 739, A. M. Lokman, T. Yamanaka, P. Lévy, K. Chen, and S. Koyama, Eds. Singapore: Springer Singapore, 2018, pp. 64–72. doi: 10.1007/978-981-10-8612-0_8.
- [5] P. Malo, A. Sinha, P. Korhonen, J. Wallenius, and P. Takala, "Good debt or bad debt: Detecting semantic orientations in economic texts," *J. Assoc. Inf. Sci. Technol.*, vol. 65, no. 4, pp. 782–796, 2014, doi: 10.1002/asi.23062.
- [6] S. Taj, B. B. Shaikh, and A. Fatemah Meghji, "Sentiment Analysis of News Articles: A Lexicon based Approach," in *2019 2nd International Conference on Computing, Mathematics and Engineering Technologies (iCoMET)*, Jan. 2019, pp. 1–5. doi: 10.1109/ICOMET.2019.8673428.
- [7] P. R. Nagarajan, M. T, S. R. A. M. Hari, K. K, and M. G, "Certain Investigation On Cause Analysis Of Accuracy Metrics In Sentimental Analysis On News Articles," in *2021 5th*

- International Conference on Electronics, Communication and Aerospace Technology (ICECA)*, Dec. 2021, pp. 1033–1037. doi: 10.1109/ICECA52323.2021.9675846.
- [8] A. Agarwal, “Sentiment Analysis of Financial News,” in *2020 12th International Conference on Computational Intelligence and Communication Networks (CICN)*, Sep. 2020, pp. 312–315. doi: 10.1109/CICN49253.2020.9242579.
- [9] W. van Atteveldt, M. A. C. G. van der Velden, and M. Boukes, “The Validity of Sentiment Analysis: Comparing Manual Annotation, Crowd-Coding, Dictionary Approaches, and Machine Learning Algorithms,” *Commun. Methods Meas.*, vol. 15, no. 2, pp. 121–140, Apr. 2021, doi: 10.1080/19312458.2020.1869198.
- [10] J. Nothman, H. Qin, and R. Yurchak, “Stop Word Lists in Free Open-source Software Packages,” in *Proceedings of Workshop for NLP Open Source Software (NLP-OSS)*, 2018, pp. 7–12. doi: 10.18653/v1/W18-2502.
- [11] A. M. Priyatno, M. M. Muttaqi, F. Syuhada, and A. Z. Arifin, “Deteksi bot spammer twitter berbasis time interval entropy dan global vectors for word representations tweet’s hashtag,” *Regist. J. Ilm. Teknol. Sist. Inf.*, vol. 5, no. 1, pp. 37–46, Jan. 2019, doi: 10.26594/register.v5i1.1382.
- [12] R. Aulianita, L. Utami, N. Musyaffa, G. Wijaya, A. Mukhayaroh, and A. Yoraeni, “Sentiment Analysis Review Of Smartphones With Artificial Intelligent Camera Technology Using Naive Bayes and n-gram Character Selection,” *J. Phys. Conf. Ser.*, vol. 1641, no. 1, p. 012076, Nov. 2020, doi: 10.1088/1742-6596/1641/1/012076.
- [13] M. A. P. Subali and C. Fatichah, “Kombinasi Metode Rule-Based dan N-Gram Stemming untuk Mengenali Stemmer Bahasa Bali,” *J. Teknol. Inf. dan Ilmu Komput.*, vol. 6, no. 2, p. 219, 2019, doi: 10.25126/jtiik.2019621105.
- [14] C. D. Manning, P. Raghavan, and H. Schütze, “Text classification and Naive Bayes,” in *Introduction to Information Retrieval*, no. c, Cambridge University Press, 2008, pp. 234–265. doi: 10.1017/CBO9780511809071.014.
- [15] “sklearn.naive_bayes.MultinomialNB — scikit-learn 1.1.1 documentation.” https://scikit-learn.org/stable/modules/generated/sklearn.naive_bayes.MultinomialNB.html (accessed Jul. 30, 2022).
- [16] J. L. Morales and J. Nocedal, “Remark on ‘algorithm 778: L-BFGS-B: Fortran subroutines for large-scale bound constrained optimization,’” *ACM Trans. Math. Softw.*, vol. 38, no. 1, pp. 1–4, Nov. 2011, doi: 10.1145/2049662.2049669.
- [17] “sklearn.linear_model.LogisticRegression — scikit-learn 1.1.1 documentation.” https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html (accessed Jul. 30, 2022).
- [18] C.-C. Chang and C.-J. Lin, “LIBSVM: A library for support vector machines,” *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, pp. 1–27, Apr. 2011, doi: 10.1145/1961189.1961199.
- [19] J. Platt and others, “Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods,” *Adv. large margin Classif.*, vol. 10, no. 3, pp. 61–74, 1999.
- [20] “sklearn.svm.SVC — scikit-learn 1.1.1 documentation.” <https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html> (accessed Jul. 30, 2022).
- [21] X. Glorot and Y. Bengio, “Understanding the difficulty of training deep feedforward neural networks,” *J. Mach. Learn. Res.*, vol. 9, pp. 249–256, 2010.
- [22] K. He, X. Zhang, S. Ren, and J. Sun, “Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification,” in *2015 IEEE International Conference on Computer Vision (ICCV)*, Dec. 2015, vol. 2015 Inter, pp. 1026–1034. doi: 10.1109/ICCV.2015.123.
- [23] D. P. Kingma and J. Ba, “Adam: A Method for Stochastic Optimization,” *3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc.*, pp. 1–15, Dec. 2014, [Online]. Available: <http://arxiv.org/abs/1412.6980>
- [24] “sklearn.neural_network.MLPClassifier — scikit-learn 1.1.1 documentation.” https://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPClassifier.html (accessed Jul. 30, 2022).
- [25] B. Zadrozny and C. Elkan, “Transforming classifier scores into accurate multiclass probability estimates,” in *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '02*, 2002, p. 694. doi: 10.1145/775047.775151.
- [26] “sklearn.linear_model.SGDClassifier — scikit-learn 1.1.1 documentation.” https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.SGDClassifier.html (accessed Jul. 30, 2022).
- [27] A. Cutler, D. R. Cutler, and J. R. Stevens, “Random forests,” in *Ensemble Machine Learning: Methods and Applications*, Boston, MA: Springer, Boston, MA, 2012, pp. 157–175.

- doi: 10.1007/9781441993267_5.
- [28] “Random forests - classification description.” https://www.stat.berkeley.edu/~breiman/RandomForests/cc_home.htm (accessed Jul. 30, 2022).
- [29] A. M. Priyatno, F. M. Putra, P. Cholidhazia, and L. Ningsih, “Combination of extraction features based on texture and colour feature for beef and pork classification,” *J. Phys. Conf. Ser.*, vol. 1563, no. 1, p. 012007, Jun. 2020, doi: 10.1088/1742-6596/1563/1/012007.