Automatic Detection of Argument Components in Text Using Multinomial Nave Bayes Classifier

Hifdzon Nur Rohman and Ibnu Asror
Data Mining Center Laboratory, School of Computing, Telkom University, Jl.
Telekomunikasi No.1, Bandung, Indonesia 40257

E-mail: mhifdzon123@gmail.com, iasror@telkomuniversity.ac.id

Abstract. Arguments are often found in various text data, for example in news, essays and articles. Argumentation Mining is a method that automatically identifies argument structures in text documents. This argument structure consists of several components that are very useful for information retrieval and processing information. In this study, a model will be built to automatically detect the component of argument, by using naive bayes classifier multinomial, the model will classify argument components into two classes, namely claim and premise. The evaluation uses k-fold cross validation. The most optimal result of this study is the average accuracy of 70.39 % and the average f1-score of 80.42 % with feature extraction, preprocessing and weighting words.

1. Introduction
Text data is the data included on natural language which is unstructured and high-dimensional, therefore, one of the benefits for text processing itself is processing the argumentation text. argumentation is a set of statements that is contained in a sentence or paragraph then usually used to persuade someone or give a reason to accept a conclusion. the component of the argument at least has one claim that is supported or rejected by at least one premise [1]. The claim is the main component in an argument, usually contains a controversial statement that the truth is not known and should not be accepted by the reader without additional support. The premise is to support the validity of the claim, premise is the reason for persuade the reader who read the claim. The argument consists of several components that show a structure based on argumentative relationships between its component [2].
Remembering the characteristic of text data which is unstructured and emphasizes on meaning, so the use of natural language processing in text data becomes very important. Because by processing the existing text, then there will be a new information or new knowledge. In this study, the detection of argument components in text data will be useful for determining information, whether it has an argument component or not. The examples of text arguments themselves can be an essays, a reviews, a news and other information providers.

In this study, the text data will be classified into two classes, claim and premise, by training the classifier first, the multinomial naive bayes classifier. The dataset is an English-language dataset that contains 90 persuasive essays which is the claim and the premise has been annotated [1]. This dataset were manually labeled for its class claim and premise, then generated 1520 data for both class. The evaluation for this research is how the impact of preprocessing for classification performance by using research validation k-fold cross validation.

There were several previous studies, one of them is the study from Palau which detects arguments on legal text [3], then study from Cabrio which identify attack and support on the online debates[4]. In this study, the classification using multinomial naive bayes classifier will be used as the method to classify the annotated arguments component. The model has been built will training the data first, then the model will automatically detecting the claim and the premise on the given data.

2. Related Work
In the study from Palau et al. which detects arguments on legal text, that is including classification problems. The classifier is trained for annotated arguments, and uses several sets of evaluated features that involve lexical, syntactic and semantic. The research focuses on automatically classify arguments on legal text / law text based on rhetorical types and visualization for easy access [3]. Then the study from Cabrio et al. focuses on the relationship between arguments and to identify what the reader can accept in online debates. the research applies the identification of the relationship of support and attack between arguments, and utilizes the result for identify the accepted argument [4].

From the related studies above, in this study, the annotated persuasive essays will be taken [1]. The method which is used for this study is the classification method of multinomial naive bayes.

3. Proposed Schema
The system that will be built on this study is automatic classification system on persuasive essays that will be done in a few step. The first step is to preprocessing the dataset, which will be analyzed for the classification results and how the preprocessing will be affected. Then the classification process will use multinomial naive bayes classifier to determine the probability of the class (premise and claim). Then for evaluating data were using k-fold cross validation to get optimal results. The following picture is the description of the system that will be built:
3.1. Dataset
The dataset is an object that represents data, and the form of this data is unstructured text, which is it will be converted into structured text to facilitate classification of the system. The dataset is 1520 records of manual labeling data from an annotation result of 90 persuasive essays made by Stab et al. with the major claim will be considered as a claim [1]. The data record is manually labeled according to the argument sentence which is the label 0 is premise and label 1 is claim, and the used attribute is the attribute of the manual labeling results, including 1,010 sentences label 0, and 510 sentences label 1.

To do the research validation, k-fold cross validation is done for data distribution with input k = 5 and k = 10 to compare the optimization. Furthermore, the dataset will be preprocessed to facilitate the system in classifying and knowing how it affects the classification results. The following table is an overview of the dataset:

| Label | Argument                                      |
|-------|-----------------------------------------------|
| 1     | Competition can effectively promote the development of economy |
| 0     | The Winner is the athlete but the success belongs to the whole team |
| 1     | They are able to sustain their cultural identities |
| 0     | The one will learn living without depending on anyone else |

3.1.1. Preprocessing In this process, datasets that have been divided into data training and data testing will be processed to get better data for the next process. Preprocessing data is a process to make low-quality data turn into high-quality data so that is easy to process [7]. Preprocessing that is conducted in this research is reducing the dimensions of the dataset, case folding, remove punctuation, tokenization, stop word removal, stemming and lemmatization. Reducing the dimensions of the dataset is the selection of dimensions in the dataset, in this case the argument.
3.1.2. **Case Folding and Remove Punctuation.** Case folding is the process of converting capital letters which is contained in datasets to lowercase letters across all datasets, and remove punctuation is the process of eliminating characters / punctuation.

3.1.3. **Tokenization.** Tokenization is the process to splitting the text corpus into individual elements that serves as input for various natural language processing algorithms [8].

3.1.4. **Stop Word Removal.** Stop Word Removal is the process of removing words which is not descriptive or not important, for example: so, or, and, the. There is a database containing a collection of stop words that usually English-language, if in the results of tokenization there words that include in database, then the word will be discarded [6].

3.1.5. **Stemming.** Stemming or porter stemmer is the process of changing the form of a word into a basic word, or step to find the root of the word for each word[6]. Each word affixes will be used as a base word, and in hopes will optimize classification performance. Example: fishing, fisher, fished becomes fish.

3.1.6. **Lemmatization.** Lemmatization is the stemming process is based on a dictionary, using the vocabulary from words to eliminate additions and return to the basic form of the word [6]. Example: see, saw, seen becomes see. The following table is the overview of the preprocessing results:

| Argument | Preprocessing Results |
|----------|-----------------------|
| Competition can effectively promote the development of economy | Competit, effect, promote, develop, economi |
| We should attach more importance to cooperation | Attach, import, cooper |

3.2. **Bag-of-Words**

Bag-of-words is simply a way of extracting features from text to be used in modeling, as in machine learning algorithms [9]. The approach is very simple and flexible, can be used in various ways to extract features from documents. Bag-of-words itself is a text representation that describes an event in words on document by ignoring the word order. The steps to get the bag-of-words are, first is to collect a collection of data in text form, then make the vocabulary from all the sentences, then calculating the the words that appear on the document. For example: Sentence 1: "you should come to my house". Sentence 2 "because my house is very clean". then the dictionary will contains a tokens ("you", "should", "come", "to", "my", "house", "because", "is", "very", "clean").

3.3. **Laplace Smoothing**

Laplace smoothing is a technique that aims to add the value of 1 (one) or other small numbers to a feature, in purpose to avoid zero probability [10]. So the way it works is by adding value 1 to each probability of the occurrence of a word that is 0 (zero), in order to avoid the fail process.

3.4. **K-Fold Cross Validation**

K-fold cross validation is a cross validation process by dividing the data that has been labeled into two parts, for training and for testing [7]. Usually 10 K is chosen, 1 K is used for testing, and the rest (K - 1) us used for training. This approach will be repeated by selecting each of the different K segments in the data as testing data, then the average accuracy of the testing is recorded.
3.5. **Classification Multinominal Naïve Bayes Classifier**

Multinomial Naive Bayes Classifier includes supervised learning based in bayes theorem for conditional probability which focuses on text classification [7]. In the general case of documents that have large sizes, this multinomial algorithm will be more effective. Here are the probability calculations based on the Bayes theorem, can be seen in the formula:

\[
P(c|x) = \arg \max_{c \in C} P(x_1, x_2, ..., x_n | C) P(c)\]

\[
P(c|x) = \arg \max_{c \in C} P(c_j) \prod_{x \in X} P(x | c)
\]

\(P(x|c)\) is a conditional probability from the words x that appears in the sentence in class c, or it can be called a probability x in class c. then \(P(c)\) is the prior probability of the sentence that appears in class c. Then the calculation will determine the placement of the class by comparing \(P(c|x)\). The prior probability calculations can be seen in the formula:

\[
\bar{P}(c) = \frac{N_c}{N}
\]

\(N_c\) is the number of sentences in class c, while \(N\) is the number of sentences from all classes. The Conditional Probability calculations can be seen in the formula:

\[
\bar{P}(w_i|c_j) = \frac{\text{count}(w_i|c_j)}{\sum_{w \in V} \text{count}(w_i|c_j)}
\]

\(\text{count}(w_i,c_j)\) is the number of occurrences of the word w in the sentence in class c, and \(w \in V\). \(\text{count}(w_i,c_j)\) is the total number of occurrences words in class c.

3.6. **Confusion Matrix**

Confusion matrix contains information about actual and predicted by classifier models [11]. The Following table are confusion matrix:

|               | Predicted |
|---------------|-----------|
|               | Negative  | Positive |
| Actual        |           |          |
| Negative      | TN        | FP       |
| Positive      | FN        | TP       |

By using confusion matrix, the measurement is more accurate than the usual accuracy (correct amount of data / total data * 100%), because in the confusion matrix it will calculating the precision.
and the recall too. Precision is to measure how much the positives predicted data is correct, the recall is to measure how much positives data is correctly predicted [11]. After calculating TN, FN, FP and TP, here the formula to evaluate the performance:

\[
\text{Precision} = \frac{TP}{TP + FP}
\] (5)

Recall Formula:

\[
\text{Recall} = \frac{TP}{TP + FN}
\] (6)

F1-Score Formula:

\[
F1Score = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}
\] (7)

4. Experiment and Analysis
In this study, the goal will be achieved by doing the classification of components in the argument sentence for the premise label and claim label, there are 3 (three) test scenarios to be used as a comparison model evaluation, there is:

| No. | Scenario | Goal |
|-----|----------|------|
| 1   | Calculates the performance of the model using bag-of-words feature extraction and evaluating k-fold cross validation (with k = 5 and k = 10) | Knowing a more optimal comparison of K input on K-fold cross validation. |
| 2   | Calculate the performance of the model using laplace smoothing and didn’t using the lapalace smoothing. | Detects the effect of the laplace smoothing for performance of the classification. |
| 3   | Calculate the performance of the model by using several preprocessing combinations | Knowing the effect of the preprocessing on the performance for the model. |

This Experiments are carried out in several stages, the first stage is splitting the data into two parts, training data and testing data, by using K-fold cross validation. Next stage is to preprocessing the dataset, after getting the preprocessing results, feature extraction bag-of- words will be used. Then the results if the feature are classified by the model that has been built, that is multinomial naive bayes classifier. And the final stage is to evaluating k-fold to estimate the performance using the confusion matrix calculation for the classification prediction. The following table is the results of scenario testing 1 in chart form:
Based on the picture above, the use of the feature extraction and K-fold cross validation with $K = 10$ is more accurate than $K = 5$. It can be seen in $K = 5$ produces an average accuracy of 64.14% and average f1-score of 74.49% while $K = 10$ produces an average accuracy of 64.93% and average f1-score of 74.99%. That is because the lower $K$ tends to be more biased than the higher $K$, and the higher $K$ can also be more effective for data with large variability. The following table is the results of scenario testing 2 in chart form:

![Chart Scenario 1](image1.png)

**Figure 2.** Chart Scenario 1.

Based on the picture above, using the laplace smoothing method will improve the performance of the model. It can be seen at $K = 10$ and using the laplace smoothing method will produce an average accuracy of 69.14% and average f1-score of 79.99% compared to without the laplace smoothing.

![Chart Scenario 2](image2.png)

**Figure 3.** Chart Scenario 2.
method will produce an average accuracy of 64.93% and f1-score of 74.99%. That is because the laplace smoothing method goals is to avoid zero probability, with the addition of 1 (one) value to each zero probability, it will be more effective and affect the performance of the model. The following table is the results of scenario testing 3 in chart form:

![Figure 4. Chat Scenario 3](image)

Adasdasd Based on the picture above, using preprocessing either can improve the performance of the model or reduce the performance of the model. It can be seen in the combination of preprocess remove punctuation and porter stemmer produces the optimal accuracy average of 70.39% and f1-score average of 80.42%. whereas the combination of preprocess remove punctuation and stop word removal produce the lowest accuracy average of 64.34% and f1-score average of 75.24%. By using the porter stemmer it will increase the accuracy and f1-score that is quite large than before, that is because the use of the porter stemmer will eliminate the affixes and search for the basic or root words, and that causes a reduction in features which will significantly improve the model’s performance. When using stop word removal, the accuracy and f1-score drops dramatically, that is because in the case of classifying claims and premise in arguments, words in the stop word dictionary are needed to create an argument, so if the words on stop word is discarded, it will reduce the evaluation results of the model performance.

The following table are the highest gap probability for each words in its class, (the data is taken from the 2nd fold):

| Words | Premise Class | Claim Class | P(word j premise) | P(word j claim) | Gap Probability |
|-------|---------------|-------------|-------------------|-----------------|-----------------|
| mobil | 13            | 6           | 0.000886          | 0.000787        | 9.8329          |
| nation| 13            | 6           | 0.000886          | 0.000787        | 9.8329          |
| must  | 14            | 8           | 0.000954          | 0.001050        | 9.6083          |
| one   | 38            | 19          | 0.002589          | 0.002494        | 9.5546          |
| cours | 11            | 5           | 0.000749          | 0.000656        | 9.3301          |
| limit | 11            | 5           | 0.000749          | 0.000656        | 9.3301          |
The table above is the gap probability for each word, where data is taken from the 2nd fold. Fold-2 is used because it has the highest f1-score, so its accuracy is guaranteed when compared to other folds. It can be seen in the gap, example for the word "nation" have a high probability of entering the premise class. So every data or sentence that has the appearance of the word "nation", will not entering the claim class, its because of the characteristic of the component premise itself. Claim is a controversial statement that is not easily accepted by the readers, unless there is a supporting sentence to support the statement. And the premise is the validity for the claim statement. The formula for gap probability is the highest probability in premise or claim class - the lowest probability in premise or claim class. So it proves that, if the probability gap gets bigger, it will cause the word chosen belongs to which classes. If the probability is highest in the premise class, the word will be often found in the premise sentence. in this study, the detection of the components of the premise and the claims on the argument are strongly influenced by preprocessing, because with the use of preprocessing the data will be more structured and easily processed.

5. Experiment and Analysis
Based on the results of the Experiments and analysis that has been done, some conclusions can be drawn as follows:

i. Each test scenario will produce different result on its performance, because on all test scenarios above can affect the performance of the model that has been built, whether it increases or decreases the performance of the model.

ii. The system has been built were successfully classifies the components claims and premise in the argument sentence, with the most optimal accuracy results of 70.39% and f1-score results of 80.42% by using the bag-of-words feature extraction, then K input by 10, laplace smoothing, and the combination of the preprocess remove punctuation and porter stemmer.

iii. The use of preprocessing were greatly affects the performance of the model, because it can increase or decrease the performance of the model. It depending on the case in the study, in the case of the argument components, the preprocess stop word removal will reduce the performance of the model dramatically.

References
[1] Christian Stab and Iryna Gurevych, Annotating Argument Components and Relations in Persuasive Essays, pp. 1501-1510, 2014
[2] Andreas Peldszus and Manfred Stede, From argument diagrams to argumentation mining in texts: a survey, pp. 1-31, 2013
[3] Marie-Francine Moens and Raquel Mochales Palau, Automatic Detection of Arguments in Legal Texts, pp. 2007
[4] Elena Cabrio and Serena Villata, Natural language arguments: A combined approach, pp. 205-210,2012.
[5] Malifauzi, Text Pre-Processing [online accessed 23 August 2018] [available : malifauzi.lecture.ub.ac.id/files/2016/02/Text-Pre-Processing-v2.pptx], 2016
[6] Charu C. Aggarwal, Data Mining The Textbook, book. 1-746, 2015
[7] Sebastian Raschka, Naive Bayes and Text Classification I: Introduction and Theory [online accessed 1 August 2018] [available: https://arxiv.org/pdf/1410.5329.pdf], 2014

[8] Jason Brownlee, A Gentle Introduction to the Bag-of-Words Model [online accessed 12 August 2018] [available: https://machinelearningmastery.com/gentle-introduction-bag-words-model/], 2017

[9] P. Goyal, What is Laplacian smoothing and why do we need it in a Naive Bayes classifier? [online accessed 23 August 2018] [available: https://www.quora.com/What-is-Laplacian-smoothing-and-why-do-we-need-it-in-a-Naive-Bayes-classifier], 2017

[10] Standford.edu, Text Classification and Naive Bayes [online accessed 1 August 2018] [available: https://web.stanford.edu/class/cs124/lec/naivebayes.pdf], 2011

[11] David M. W., EVALUATION: FROM PRECISION, RECALL AND F-MEASURE TO ROC, INFORMEDNESS, MARKEDNESS CORRELATION, pp. 37-63, 2011