CMPE251 PROJECT REPORT

Andrew Sheldon
20026539
16aws1@queensu.ca

NOVEMBER 27, 2020

QUEEN’S UNIVERSITY

Introduction

This report presents predictors of U.S. presidential elections based on word usage by the candidates running for office. The analyses performed for this report were performed using KNIME. Screenshots of the workflows used are provided for all findings that follow. Decision trees used to determine words that have a significant effect on election outcomes can be found in the appendix. The words that appear in these decision trees are words that the predictor deems influential to the outcome. These words are discussed, along with potential reasons for their influence.

Hypothesis

The collection of the 1000 most commonly used words in candidate speeches will likely be a very noisy dataset. KNIME will likely draw some apparent conclusions from this data, but these may tell us more about the speech patterns of successful candidates than attributes which actually make them successful. Examining words like conjunctions and determiners will likely tell us less about how successful candidates construct their sentences than a direct analysis of the speeches themselves (although it may give some indication into what to look for). Still, the noise provided by these types of words may be too great to provide any proper insights.

We should expect to receive better information from the dataset of deceptive words. This data is cleaner, being limited to words that experts think might actually have a direct effect. Correlations found in this data are more likely to be meaningful, and we expect to find more of them.

It is worth noting that despite the size of this dataset, the data only encompasses the speeches of three successful presidential candidates: Bill Clinton, George W. Bush, and Barack Obama. In the context of data analytics, this is an extremely small sample size. The ideal way to expand the sample size is debateable, however. Winners of down-ballot congressional races may provide little information, as few people listen to these speeches, letalone would be swayed by what they hear. Election winners in other countries do not necessarily possess attributes that the American electorate would look for. Searching further back in time for data may be ineffective because speech patterns and charismatic attributes change over time.

Modeling & Analysis

Reading confusion matrices

Models in this report are evaluated based on the accuracies calculated from their respective confusion matrices. Confusion matrices are tables used to demonstrate the predictor's output and compare its predictions to the real reserved for the test set. Entries along the diagonal denote the number of times the predictor was correct. The off-diagonal entries denote cases where the predictor was incorrect: in this case, either false negatives or false positives. Accuracy is computed by dividing the number of correct predictions out of the total number of predictions i.e. the size of the test set.

Predicting a winner based on word usage

The following workflow was used to accrue the data found in this section, where the learner and predictor blocks were swapped out based on the model being assessed. The binner was added so that models would have the option to treat similar values as equal, if the learner found that to be a more effective strategy. Note that binned data does not replace the original data – both sets are available to the model.

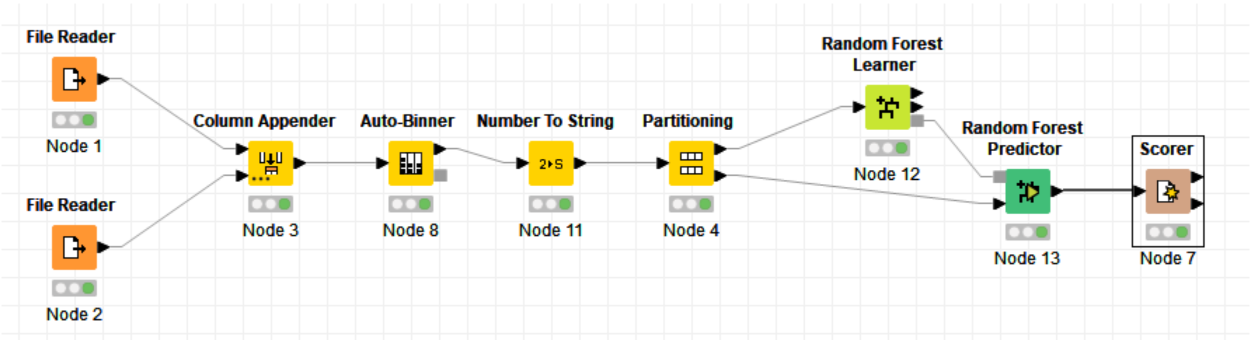

Figure 1: KNIME workflow layout used in this section.

Given that the purpose of this project is to find words whose usage influences a candidate's chance of winning an election, clustering methods were used to analyze the data. In theory, we want to categorize words based on the positive or negative effect they have on the electoral outcome.

The first analysis was performed using a decision tree model. This learner can measure the quality of its model using either Gain Ratio or GINI index. The decision tree learner yielded the following respective confusion matrices using these attributes.

Table 1: Confusion matrices generated by decision tree learner. Left: decision tree learner using gain ratio as distance metric. Right: decision tree learner using GINI index as distance metric.

RowID	1	0
1	70	22
0	8	30

RowID	1	0
1	63	23
0	14	22

Using Gain Ratio, the decision tree model was able to predict the correct winner of the election with an accuracy of 76.9%. Using GINI index, the predictor had an accuracy of only 69.7%. Allowing MDL pruning by the learner yielded slightly better accuracies of 84.6% using the gain ratio metric and 74.6% for GINI index. These respective confusion matrices are shown below.

Table 2: Confusion matrices generated by decision tree learner with MDL pruning. Left: decision tree learner using gain ratio as distance metric. Right: decision tree learner using GINI index as distance metric.

RowID	1	0
1	82	10
0	10	28

RowID	1	0
1	75	17
0	16	22

The next algorithm examined was the random forest predictor. Being an aggregation of decision tree models, it ended up yielding a slightly better accuracy of 87.7%. The confusion matrix for this model is shown below.

Table 3: Confusion matrix generated by random forest predictor.

RowID	1	0
1	86	6
0	10	28

Tree ensembles were also tried, resulting in an accuracy of 84.6% and the following confusion matrix.

Table 4: Confusion matrix generated by tree ensemble predictor.

RowID	1	0
1	84	8
0	12	26

The last clustering algorithm explored is the self-organizing tree algorithm. It was by far the most computationally expensive model used in this section, but did sport an impressive 86.2% accuracy using both Euclidian and Cosinian distance metrics.

Table 5: Confusion matrices generated by SOTA predictor. Left: SOTA learner using Euclidian distance as quality metric. Right: SOTA learner using Cosinius distance as quality metric.

RowID	1	0
1	90	2
0	16	22

RowID	1	0
1	83	9
0	9	29

It is interesting to note that using Euclidian distance causes the model to predict a winner more often, whereas Cosinian distance measurement makes the model more likely to predict a loss. Despite this, they are equally biased to either side, yielding identical accuracy figures.

Next, a cross-validator was added to the model to further evaluate the previous builds which were deemed the most promising or successful. The resulting workflow is shown below.

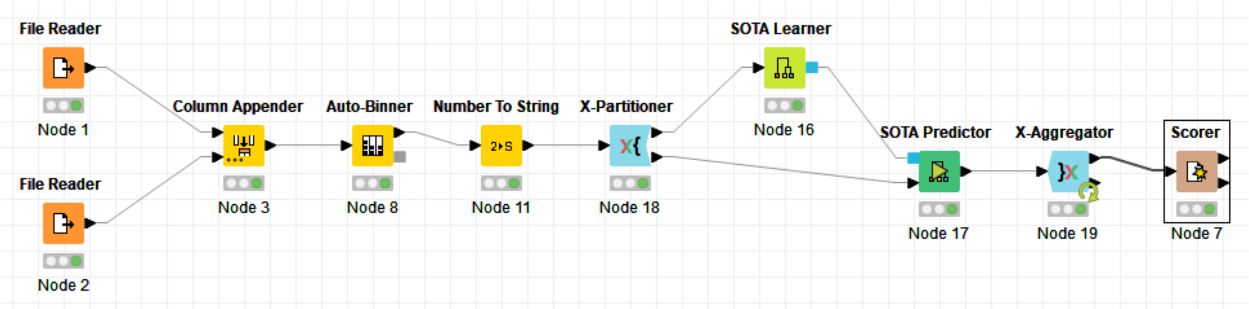

Figure 2: KNIME workflow layout used in this section.

Cross-validation was first performed on the random forest model. Unfortunately, cross-validation caused the model's accuracy to drop to 85.4%, characterized by the confusion matrix below.

Table 6: Confusion matrix generated by random forest predictor.

RowID	1	0
1	273	16
0	47	95

This drop in accuracy likely can be attributed to the random nature of the model, and the initial predictor's relatively high accuracy was likely a fluke. Subsequent trials using the original random forest workflow saw decreased accuracies of 80%, 83.1%, and 82.3%, reinforcing the idea that the first results were a fluke.

Next, cross-validation was applied to the tree ensemble model. The first attempt resulted in the same confusion matrix as the cross-validated random forest; the second attempt yielded the matrix below.

Table 7: Confusion matrix generated by tree ensemble predictor.

RowID	1	0
1	269	20
0	52	90

This has an accuracy of 83.3%, which is unsurprisingly within the same range as the random forest models.

Lastly, the SOTA model was run through the cross-validation workflow to validate its results. The following confusion matrices are for the cross-validated SOTA predictor, first using the Euclidian distance metric and then Cosinian distance.

Table 8: Confusion matrices generated by SOTA predictor. Left: SOTA learner using Euclidian distance as quality metric. Right: SOTA learner using Cosinius distance as quality metric.

RowID	1	0
1	268	21
0	41	101

RowID	1	0
1	261	28
0	44	98

These results correspond to respective accuracies of 85.6% and 83.3%. This result under Euclidian distance represents the highest accuracy of all the cross-validated models, somewhat justifying its comparatively high computation time. However, these accuracies are both less than when the workflow was executed without cross-validation.

Regression

The target data (winners.csv) only takes binary-valued data indicating the winner or loser of an election. Unfortunately we have no information about polls or win margins, so performing a regression provides little useful information. A better performance in speeches does not result in any measurable difference in the target attribute – it only influences the probability of this value being either 1 or 0.

In an ideal study, we would have data pertaining to win margins, in terms of either the popular vote or the electoral vote. This data would actually be relatively easy to acquire. In a perfect world we would also have a method of normalizing the attributes of each candidate in comparison to any other candidate, rather than just the one or two opposing candidates that they ran against. The bottom line is that none of this is available in the given dataset, so we must work with what we have. Unfortunately, this means we cannot perform a meaningful regression analysis on the data.

Predicting a winner based on use of deceptive language

As noted in the hypothesis, good presidents typically emit the aura of being versatile, skilled in many ways. They often win nominations and elections based on the impression of versatility that they give. Deceptive language, in this dataset, is defined as words that deceive the listener into believing this illusion.

The overall structure of the dataset remains the same, so the same models will be used as before. The first algorithm is the decision tree algorithm. The following confusion matrices were generated using gain ratio as the quality metric; the first table without pruning, the second using MDL pruning.

Table 9: Confusion matrices generated by decision tree predictor using gain ratio as quality metric. Left: no pruning. Right: using MDL pruning.

RowID	**1**	**0**	**RowID**	**1**	**0**
1	223	61	1	288	1
0	70	71	0	136	6

Without pruning the model attained an accuracy of 69.2%, compared to 68.2% with pruning. With pruning, note how few losses were predicted, and how many false positives were obtained. Next, under GINI index, the model produced the following confusion matrices.

Table 10: Confusion matrices generated by decision tree predictor using GINI index as quality metric. Left: no pruning. Right: using MDL pruning.

RowID	**1**	**0**	**RowID**	**1**	**0**
1	220	52	1	232	55
0	52	83	0	74	68

The table on the left represents a 74.4% accuracy, and the table on the right represents 69.9%. Interestingly, where MDL pruning increased the predictors' accuracy when dealing with the 1000 most commonly used words, it decreases the accuracy when dealing with this dataset.

Next, the random forest predictor, which previously was the most accurate model when dealing with the prior dataset of commonly used words, yielded an accuracy of 72.4%. The corresponding confusion matrix is shown below.

Table 11: Confusion matrix generated by random forest predictor.

RowID	**1**	**0**
1	245	44
0	75	67

This is a disappointing find, given the model's earlier success. Cross-validation with seed randomization yielded similar results. The tree ensemble predictor, with accuracy 71.7%, was similarly disappointing:

Table 12: Confusion matrix generated by tree ensemble predictor.

RowID	**1**	**0**
1	240	49
0	73	69

Lastly, the SOTA predictor. With cross-validation, this model attained a 69.1% accuracy using Euclidian distance as its quality metric and 67.3% using Cosinius distance. The respective corresponding confusion matrices are presented below.

Table 13: Confusion matrices generated by SOTA predictor. Left: SOTA learner using Euclidian distance as quality metric. Right: SOTA learner using Cosinius distance as quality metric.

RowID	1	0
1	220	69
0	64	78

RowID	1	0
1	217	72
0	69	73

Although substantially less accurate than when dealing with the previous dataset, these results are not necessarily negative. Each of these models was able to predict the correct election winner well over 50% of the time. This implies there likely does exist some relation between the rate at which these words are used and subsequent election outcomes.

The weaker predictor efficacy when using the deceptive language dataset is a direct contradiction to the hypothesis. This will be further explored after isolating which specific words have the greatest effect on the target attribute.

Optimal word usage

The following decision tree workflow was used to determine which words are most likely to affect the outcome of an election. Decision trees were selected for this section for two reasons. Firstly, they provided sufficient accuracy when used before, especially with MDL pruning. Secondly, they are easier to read and extract data from in KNIME. By contrast, random forests provide a collection of decision trees, of which KNIME does not provide the accuracy of any single tree. The same applies to the tree ensemble, and the SOTA model provided by KNIME is virtually illegible.

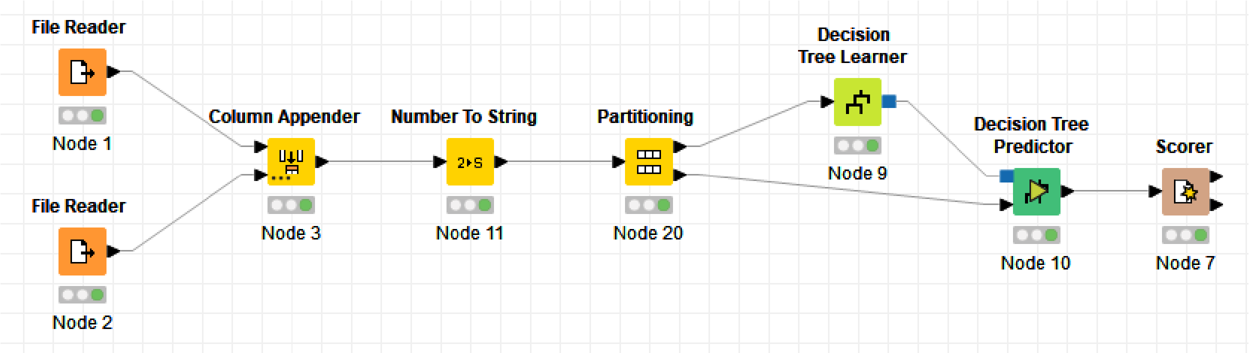

Figure 3: KNIME workflow layout used in this section.

Below is the SOTA decision tree generated by the learner. KNIME does not provide an option to label the tree.

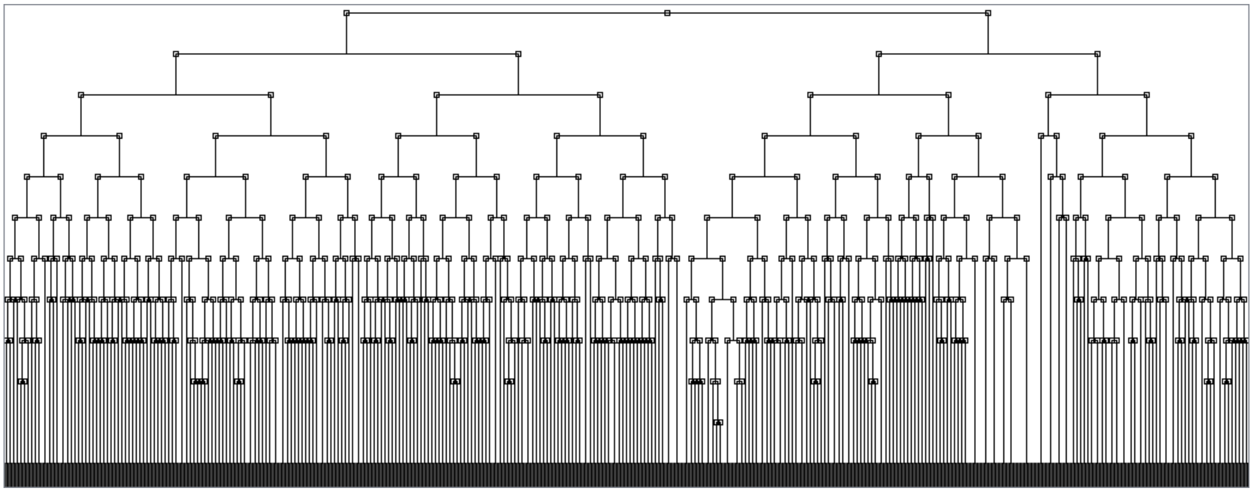

Figure 4: KNIME-generated SOTA decision tree.

Therefore, the decision tree model was decided to be the most effective tool in determining which words might influence the outcome of an election. Using the GINI index quality measure (with MDL pruning), the resulting decision tree yielded an impressive accuracy of 84.6%. The confusion matrix is provided below for reference. The decision tree that the learner built can be found in the appendix.

Table 14: Confusion matrix generated by decision tree learner using GINI index as quality measure, with MDL pruning.

RowID	1	0
1	84	6
0	14	26

The decision tree looked at the following columns of the dataset.

Table 15: Attributes examined in the decision tree found in Figure 5.

Attribute	Corresponding word	Effect of use
col117	why_WRB	Good
col228	laughter_NP	Good
col385	public_JJ	Bad
col785	main_JJS	Good
col814	union_NN	Good
col55	because_CS	Good
col930	left_VBN	Bad

Using gain ratio as the quality measure also yielded a model with 84.6% accuracy. Its decision-making process evaluated the usage rates of the following words.

Table 16: Attributes examined in the decision tree found in Figure 6.

Attribute	Corresponding word	Effect of use
col667	greater_JJR	Bad
col322	obama_NP	Bad

col939	balanced_VBN	Bad
col13	this_DT	Good
col906	worker_NN	Bad
col500	interests_NNS	Bad
col495	before_IN	Bad
col237	opportunity_NN	Bad

In both cases the removal of MDL pruning decreased the model's accuracy, hence this is the set of words most likely to affect the outcome of the election. It is interesting that when using GINI index as the model's quality measure, the model looked largely for words that increase the candidate's chance of winning, while when using gain ratio the model found a collection of words that mainly decrease the candidate's chance of winning.

Firstly, note the presence of Obama's name in the second list. Naturally, it is unlikely that Obama would refer to himself often in the third person. John McCain would have been the candidate saying his name the most of any candidate examined in this project, and McCain's loss is the likely explanation for Obama's name appearing in the 'bad' column. More interestingly, however, no other candidate's name appears to have either a positive or negative effect, according to these models. It is possible that other candidates mentioned their opponent less than McCain did. Given that McCain lost, this may be an effective strategy.

Of the remaining words, the "good" ones are why, laughter, main, union, because, and this. The first one and last two are likely to make their way into any candidate's speech. The following words are less likely to see as consistent use.

- Laughter: laughter is inherently positive. It is a light-hearted reaction innate to all humans. Laughing humanizes a person, and using the word "laughter" might give the perception of an understanding of the human spirit, and an indication that the voter can relate to the candidate.
- Main: synonyms of main include principal, chief, foremost, and focal. These words might come across as either too technical or too formal, disconnecting the candidate from the voter. Main is a simpler word than these alternatives. Furthermore, there is a chance that voters in Maine feel a connection when they hear their state named.
- Union: this is a word that can come up in many different contexts. It can refer to labour unions, which millions of voters belong to. Addressing labour unions is common practice in candidate speeches and presidential debates. In addition, the United States of America is often referred to as "the Union" and name-dropping the country in this way can be a reminder of the candidate's patriotism.

Aside from Obama's name, words that might have a negative effect on the outcome of the election are public, left, greater, balanced, worker, interests, before, and opportunity. Below are possible reasons for the inclusion of some of these words on the list.

- The word "left" most likely refers to left-wing politics and its supporters. Addressing politicians and voters in this manner can be seen as divisive. Voters are not being treated as individuals anymore, and perceive themselves as being grouped together.
- "Before" is a word that focuses on the past. Presidents are expected to be forward-thinking and lead America into the future.

- The word “worker” might be seen as demeaning, especially compared to similar words such as “employee”. No voter wants to be thought of as a mere category to be either pandered to or written off, similarly to the word “left”.
- Use of the word “opportunity” could be seen as a cop-out. People don’t want to hear about opportunities, if they are looking for action.

Next, the same analysis was performed on the deceptive language dataset. While these words may be used less often, their association with the ability to present a more multi-dimensional façade might make them more influential than most other words. Using gain ratio as the metric, the model predicted only 4 losses out of the 130 speeches in the test set - not characteristic of a good predictor. On the other hand, using the GINI index metric, the model yielded an accuracy of 77.7%. This model’s decision tree can be seen in the appendix, and its results are below.

Table 17: Attributes examined in the decision tree found in Figure 7.

Attribute	**Corresponding word**	**Effect of use**
col10	look	Good
col3	my	Bad
col6	I'm	Good
col1	but	Good
col33	however	Bad
col0	i	Bad
col12	without	Good
col5	me	Good

Perhaps the most interesting takeaway from this information is the words “I” and “my” having a negative effect on a candidate’s election prospects, but the words “me” and “I’m” having a positive effect. While these words appear similar on the surface, and one would think there would be no effective difference between them, this is a perfect example of the power of data analytics. Their interchangeability makes it easy to reword speeches using this information. The same goes for the words “but” and “however” – these words are synonymous, and it is easy to switch one out for the other. While it is possible the apparent effect of these words is mere coincidence, it costs nothing to make the change. If there is an actual effect, we can expect it to be a positive one.

We should also be wary that the classifications depicted in this table only resulted in an accuracy of 77.7% in practice. This is lower than the model that was used on the set of most commonly used words, and far from a certainty. Reintroducing the binner to the workflow did not deliver any improvement to the output accuracy in this case.

Conclusion

Based on the KNIME-generated data in this report, it may be beneficial to prioritize the use of certain words over others when writing speeches for US presidential candidates. However, not a single model tried in this report was able to predict election winners with an accuracy of over 90%.

Assuming these predictors are sufficiently effective on future datasets, it is seen that some of the most common words used in speeches are detrimental to a candidate’s election prospects. The elimination of these words from a candidate’s speeches may have some positive effect on the outcome of an election. Unfortunately, it is impossible to quantify this effect as we do not have access to data pertaining to elections where these words have been eliminated from a candidate’s speeches. As a result, it is impossible to make any verifiable conclusions from this analysis.

However, we are able to make recommendations based on these findings. For future candidates, it would be worth exploring the word substitutions outlined in this report. While there is no guarantee that these modifications would help a candidate’s chances of being elected, the probability of them having a negative effect is far less likely. Furthermore, future candidates adhering to these recommendations would provide the data necessary to either validate or refute this report’s findings.

We can say with slightly stronger confidence that the substitution of words that are found in the 1000 most commonly used words has a greater effect than the recommended substitutions of words deemed deceptive. Neither has a guaranteed effect, but both have a substantial chance of being effective.

More importantly, this project has generated predictors that can determine the winner of presidential elections based on the words publicly spoken by candidates, and their words alone. Of all these models, the results are far from guaranteed. But for applications such as betting, these models provide enough confidence to reasonably assume who the next president will be. Even better, evaluating the 1000 most common words provides more confidence than isolating a smaller set of expert-selected words that are not necessarily used often, virtually guaranteeing a sufficient sample size.

Appendix A: Decision trees

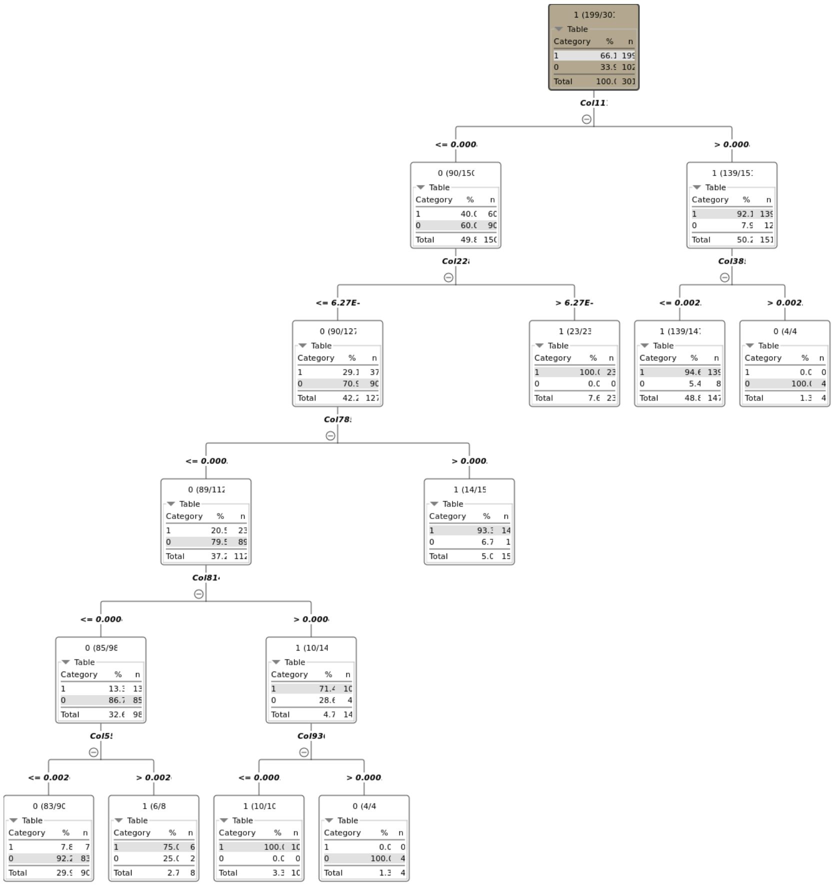

Figure 5: KNIME-generated decision tree for mostfreq1000docword.csv dataset using GINI index as quality metric.

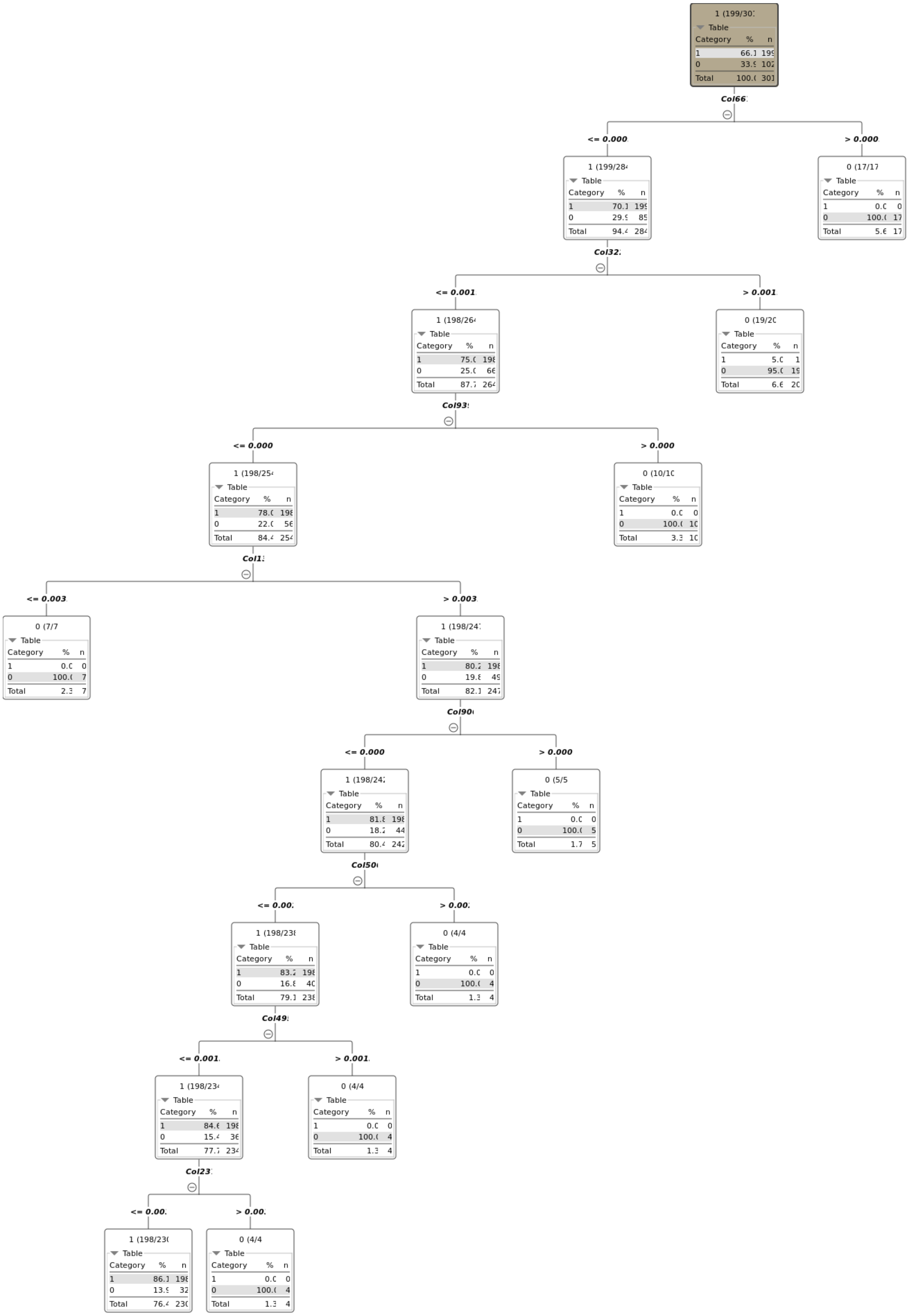

Figure 6: KNIME-generated decision tree for mostfreq1000docword.csv dataset using gain ratio as quality metric.

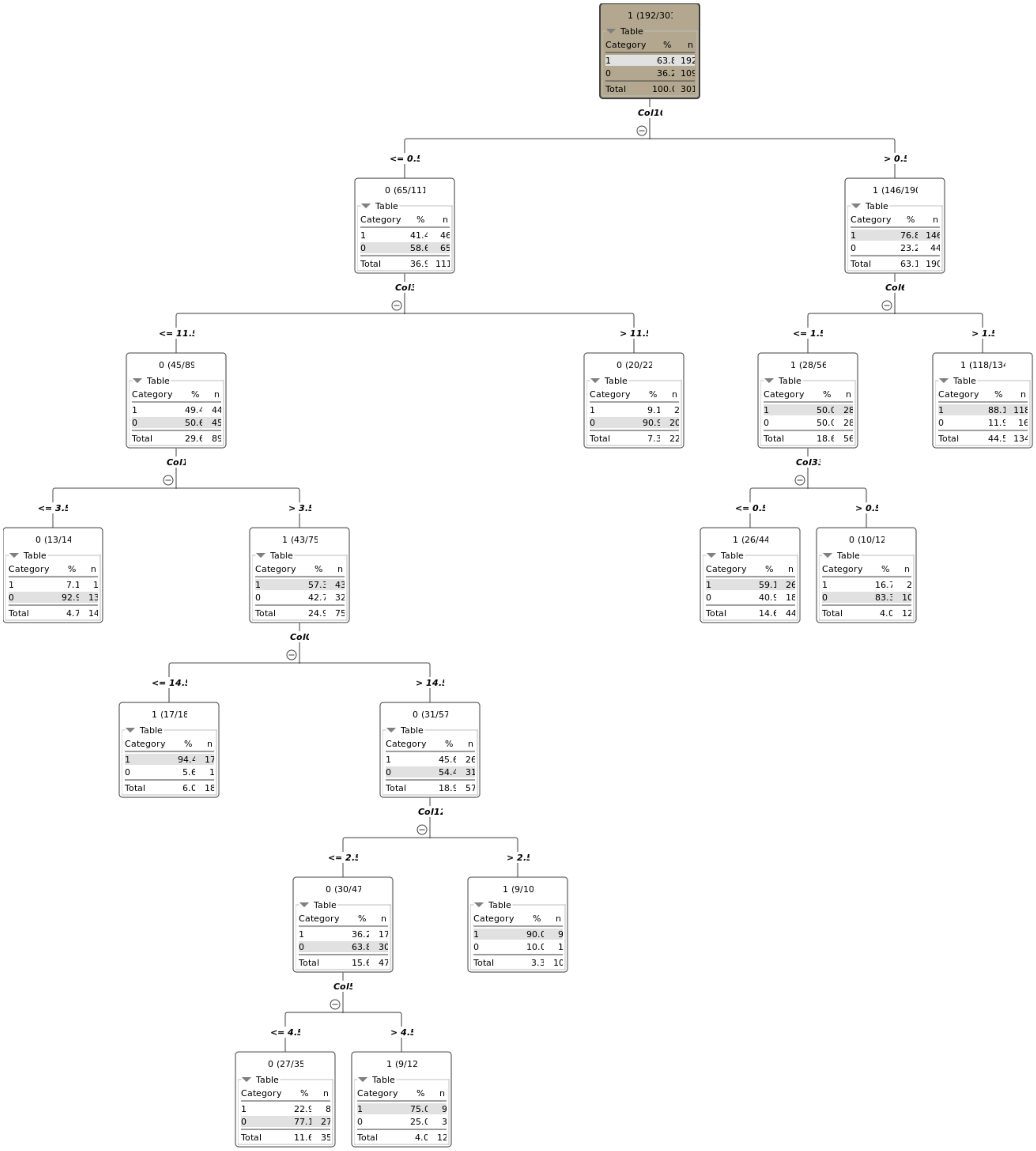

Figure 7: KNIME-generated decision tree for deceptiondocword.csv dataset using GINI index as quality metric.