Discuss one thing you notice that is different between the two emails that might relate to the identification of spam.

The formats of the first ham and the first spam are quite different. While the first ham starts with a URL and followed by clear texts, the first spam started with

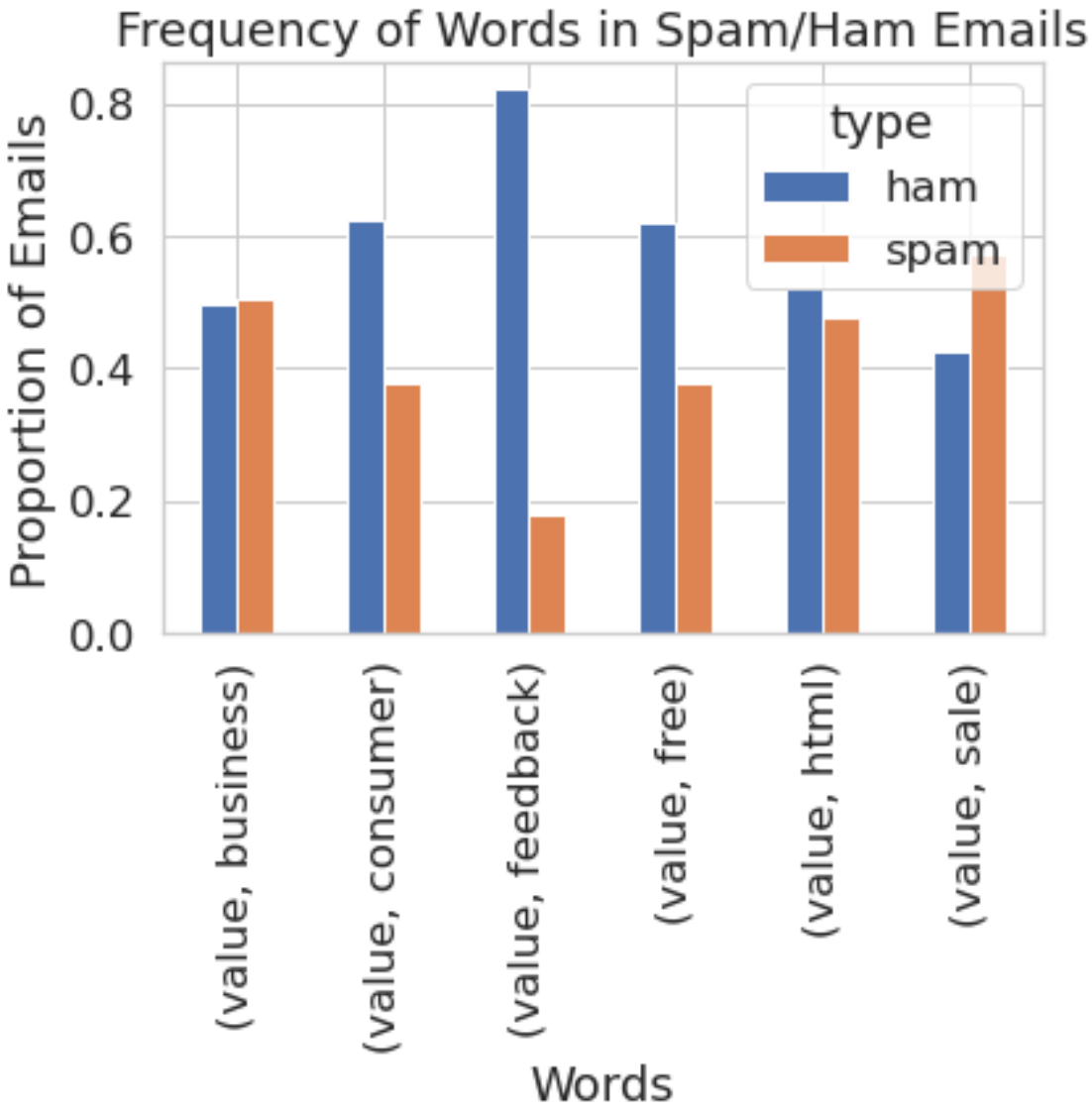
and followed by mixture of commands and words.

0.0.1 Question 3

Create a bar chart like the one above comparing the proportion of spam and ham emails containing certain words. Choose a set of words that are different from the ones above, but also have different proportions for the two classes. Make sure to only consider emails from `train`.

In [100]:
```
train = train.reset_index(drop=True) # We must do this in order to preserve the ordering of e

spam = train[train['spam'] == 1]['email']
ham = train[train['spam'] == 0]['email']
words = ['html', 'feedback', 'consumer', 'free', 'sale', 'business']
spam_df = pd.DataFrame(words_in_texts(words, spam), columns = words)
ham_df = pd.DataFrame(words_in_texts(words, ham), columns = words)

spam_df['type'] = 'spam'
ham_df['type'] = 'ham'

merged = spam_df.append(ham_df).melt('type')
merged_pivot = pd.pivot_table(merged, index = 'type', columns = 'variable', aggfunc = np.sum)
total = merged_pivot.iloc[0, [0, 1, 2, 3, 4, 5]] + merged_pivot.iloc[1, [0, 1, 2, 3, 4, 5]]
merged_pivot = merged_pivot / total

merged_pivot.T.plot(kind = 'bar')
plt.ylabel('Proportion of Emails')
plt.xlabel('Words')
plt.title('Frequency of Words in Spam/Ham Emails')
```

Out[100]:
```
Text(0.5, 1.0, 'Frequency of Words in Spam/Ham Emails')
```


Frequency of Words in Spam/Ham Emails
Proportion of Emails
0.0
0.2
0.4
0.6
0.8
type
ham
spam
(value, business)
(value, consumer)
(value, feedback)
(value, free)
(value, html)
(value, sale)
Words

0.0.2 Question 6c

Comment on the results from 6a and 6b. For **each** of FP, FN, accuracy, and recall, briefly explain why we see the result that we do.

1. FP equals to 0 because the Zero_Predictor never predicts positive, therefore there is no false positive.
2. FN equals to the sum of all positive because whenever the true value is positive, the prediction would be false.
3. The accuracy equals to the ratio of 0s in Y-train to 1s, because the Zero Predictor would always be right when the true value is 0.
4. the Recall equals to 0, because the numerator TP is always 0 due to the Zero Predictor.

0.0.3 Question 6e

Are there more false positives or false negatives when using the logistic regression classifier from Question 5?

There are more false negatives when using the logistic regression from Question 5

0.0.4 Question 6f

1. Our logistic regression classifier got 75.76% prediction accuracy (number of correct predictions / total). How does this compare with predicting 0 for every email?
2. Given the word features we gave you above, name one reason this classifier is performing poorly. Hint: Think about how prevalent these words are in the email set.
3. Which of these two classifiers would you prefer for a spam filter and why? Describe your reasoning and relate it to at least one of the evaluation metrics you have computed so far.

1. The prediction accuracy of logistic regression is close to the accuracy of the Zero Predictor, whose accuracy is 74.47%.
2. One reason that the model performs poorly might be that the word list used as featured engineering are usually used in both Spam and Ham emails. Thus, this set of words was no the best choice to distinguish spams.
3. I prefer the logistic filter. For recall, the Zero Predictor has an result of 0, while the logistic predictor has a positive result. Considering the close accuracies of the two predictors, this means that the logistic predictor does a better job at predicting.