



# Contract Analysis: Finding Which Party a Legal Contract Favors

Theo Culhane [tculhane@stanford.edu](mailto:tculhane@stanford.edu)

Stanford  
CS 221

## Overview

One issue that many people struggle with, both in their everyday and professional lives, is dealing with contracts that take away more rights than they might realize they are signing away, anywhere from the terms and conditions of their social media account to the terms on their housing loan. I wanted to try to begin solving that by helping people figure out just how “skewed” or “predatory” the contracts they are signing are.

## Model

- Score each sentence in a contract based on its “skew”
  - Negative numbers represent a skew towards the first party mentioned
  - Positive numbers represent a skew towards the second party mentioned
  - Magnitude represents how strongly the sentence creates a skew in the contract

## Sample Input/Output

“Party B may not sue party A on any grounds

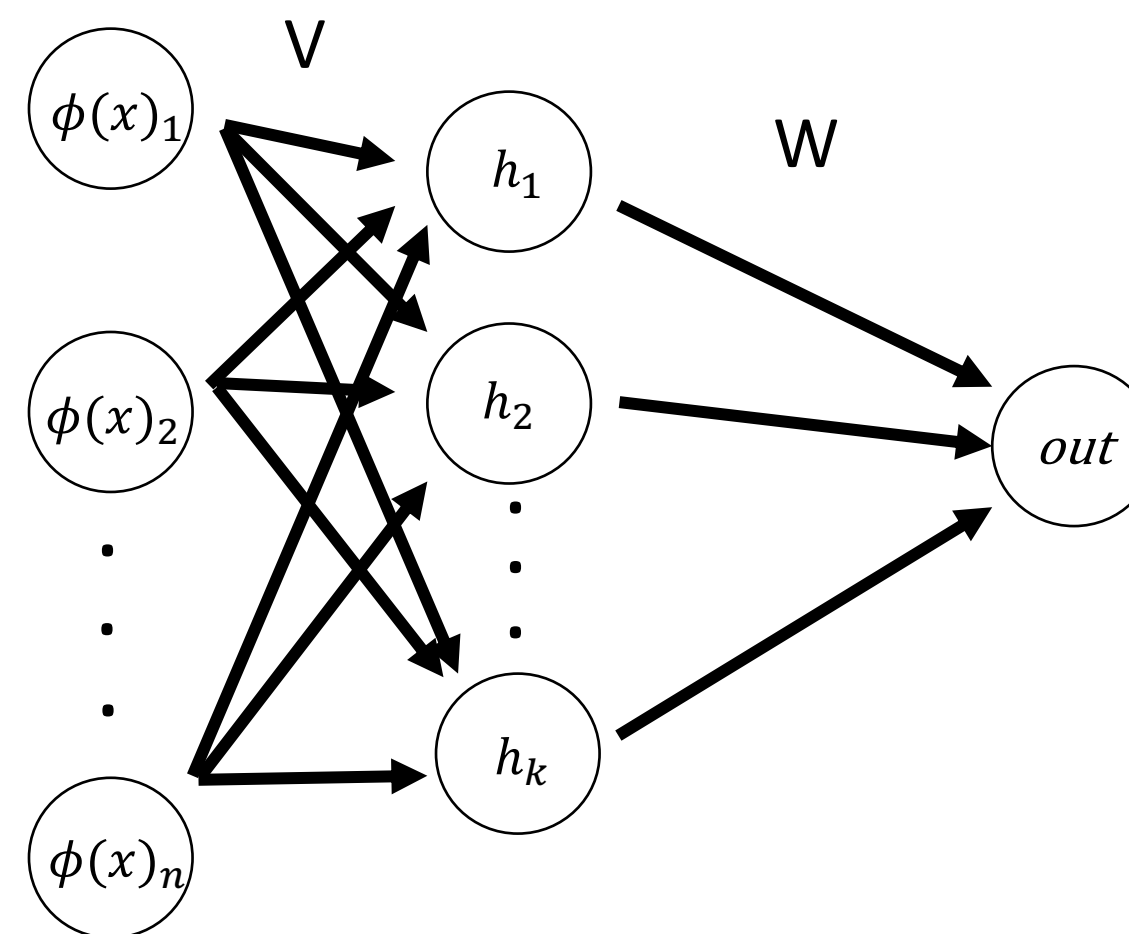
whatsoever” – Should score around a -1

“Party A will give party B \$10,000 in exchange for their car” – Should score around a 0

“Party A shall give party B all of their assets in exchange for a yo-yo and two hotdogs” – Should score around a 1

## Approach

- Use a two-layer Multi-Layer Perceptron to make predictions about what the skew of each sentence is, with  $n$  features and  $k$  hidden nodes
- Used sigmoid functions and the squared loss



## Challenges and Limitations

- Limited training data
  - Because a true analysis of a contract requires a trained lawyer’s opinion, reliable training data is time intensive and difficult to get
- Highly context sensitive
  - The same obligation, such as paying the other party a certain amount of money, can have wildly different impacts on how skewed the sentence is depending on what else the contract requires

## Feature Templates

- Number of occurrences of each word
- Indicator for each word
- Indicator for each party
- Indicator for whether party A comes before party B
- Indicator for the word directly before each party
- Indicator for the word directly after each party

## Results

Algorithm	Training Error	Test Error
Linear Predictor	0.05	0.4
Neural Net	0.007	0.05

Interesting side note: While testing using contracts all written by the same lawyer as training and test data, test error was 0

## Discussion

- Predictably, a neural net far outperformed a linear predictor for determining what made a contract biased
- This particular MLP far outperformed my expectations for performance. I had expected maybe 10% error given how little training data I was able to get, and the program got 5%
- For further exploration, I am interested in finding the optimal number of hidden nodes and testing out different features
  - I am also interested to test out a deeper MLP, with perhaps two or three hidden layers