Deception Detection

Final Project - NLP & the Web

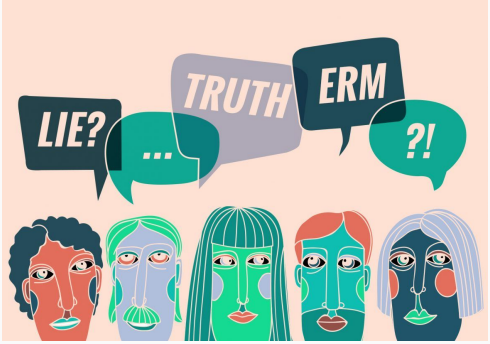
Sebastian Lembcke **Danu Caus** **Nambiar Shruti Surendrakumar** **Nana Baah**

Outline

- Introduction
- Dataset and Preprocessing
- Software Setup
- Feature Extractors
- Evaluation Results
- Data analytics and visualizations
- GUI Demo
- Conclusion and Future Work

Introduction

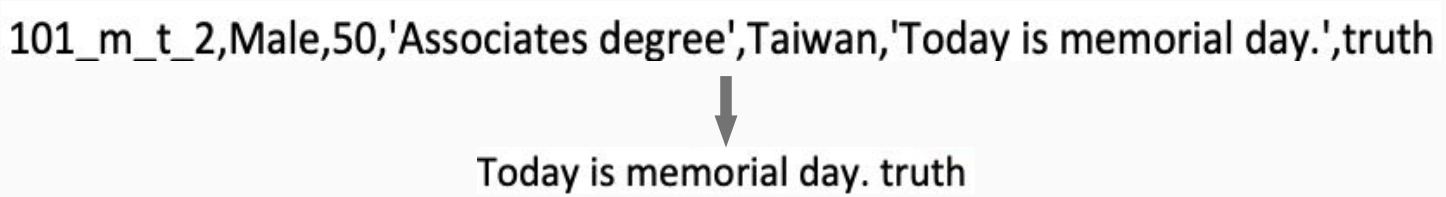
- Analyze how linguistic properties of statements can be utilized to infer if they are truthful or deceptive
- Find common syntactical patterns in lies and truths
- Determine which topics people tend to lie about and which terms are used for this purpose.



Dataset and Preprocessing

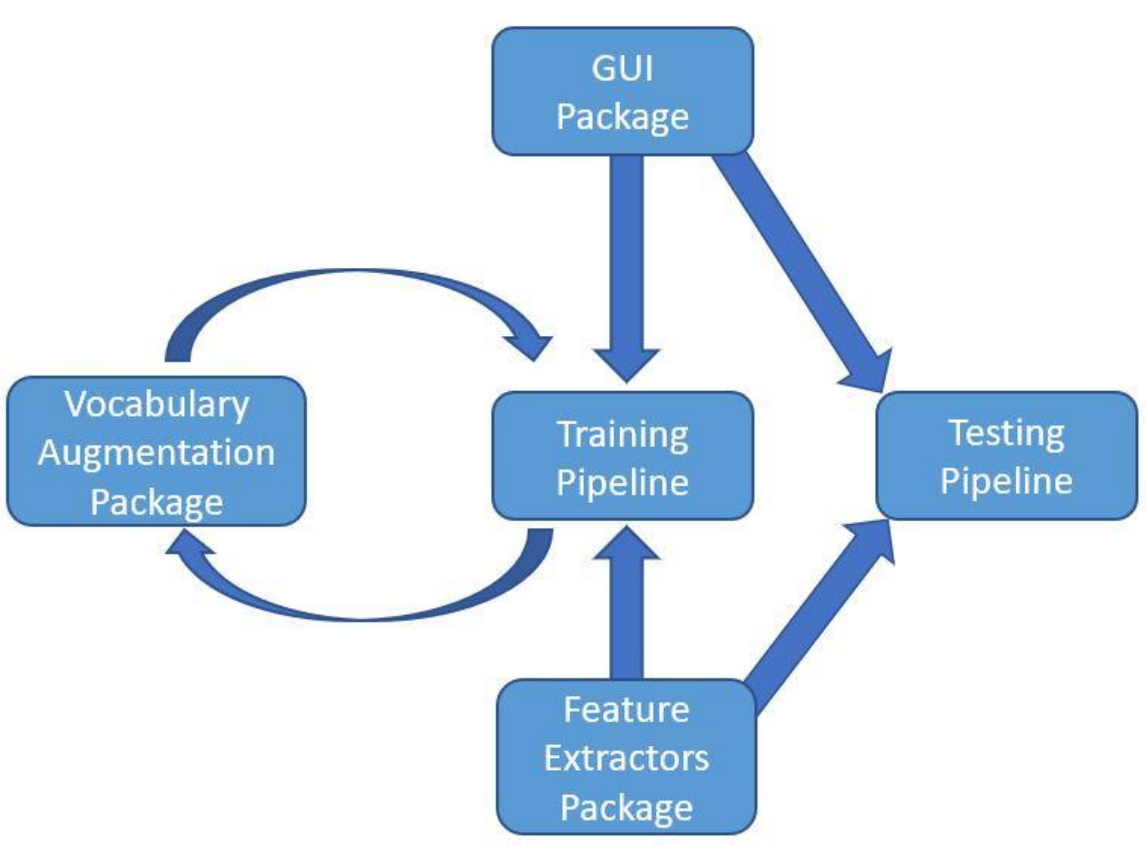
Dataset - 7 Truths 7 Lies

- Crowdsourced deception dataset consisting of short open domain truths and lies from 512 users.
- Preprocessed with python script (extract sentence and label, shuffling, 50/50 split)



Software Setup

- Main pipeline with UIMA framework and ClearTK library
- Separate Vocabulary Augmentation project using WordNet dictionary and MIT JWI API
- GUI with Java Servlet Technology
- Data Analysis using LIWC
- Visualizations using the Python Scattertext tool



Feature Extractors

Lexical Extractor

Vocabulary built from dataset by tokenizing statements.

Extractor counts the number of words labelled as truth/lie

Readability Scores

Flesch Kincaid Readability Metric

Smog Grade Readability Metric

Syntactic complexity measures: number of characters and syllables

Syntax Features

Uses Stanford CFG Parser

Extract features based on part-of-speech tags, dependency types and grammar representation tree

Other Features

Exaggeration extractor

First person pronoun extractor

Negation extractor

Syntax based Features

- From parser data extract psycholinguistic features
- When lying, people often make use of more cognitively complex constructions
- Select best performing features

Input	I made a continental breakfast for my roommates today.
Part-of-speech tagged sentence	I/PRP made/VBD a/DT continental/JJ breakfast/NN for/IN my/PRP$ roommates/NNS today/NN ./.
Parsed Tree	(S (NP (PRP I)) (VP (VBD made) (NP (DT a) (JJ continental) (NN breakfast)) (PP (IN for) (NP (PRP$ my) (NNS roommates))) (NP-TMP (NN today))) (. .))
Dependency Information	DepParseInfo{depParents={1=2, 2=0, 3=5, 4=5, 5=2, 6=8, 7=8, 8=2, 9=2, 10=2}, depLabels={1=nsubj, 2=root, 3=det, 4=amod, 5=dobj, 6=case, 7=nmod:poss, 8=nmod:for, 9=nmod:tmod, 10=punct}}

Table: Sample Input-Output when using the Stanford NLP Parser

Index	Syntax Feature Type	F1 Score
1	Number of Prepositions	0.56891
2	List dependency present	0.56714
3	Number of Conjunctions	0.56647
4	Discourse elements present	0.56647
5	Number of Adverbs	0.56597
6	Number of Verbs	0.56556
7	Exclusion words present	0.56530
8	Number of Adjectives	0.56529
9	Number of Nouns	0.56384
Combined 1-4	*Top 4 features*	*0.56849*

Table: Syntactic Features and corresponding F1 scores

Evaluation Results

Best Feature Extractor Set
Lexical Extractor
Syntax Features Extractor (parsed tree, number of prepositions, presence of lists of items, number of conjunctions, presence of discourse elements)
Readability Scores Extractor (Flesch Kincaid score, Smog grade, number of characters and number of syllables)
First person pronoun extractor
Negation Extractor
Exaggeration Extractor

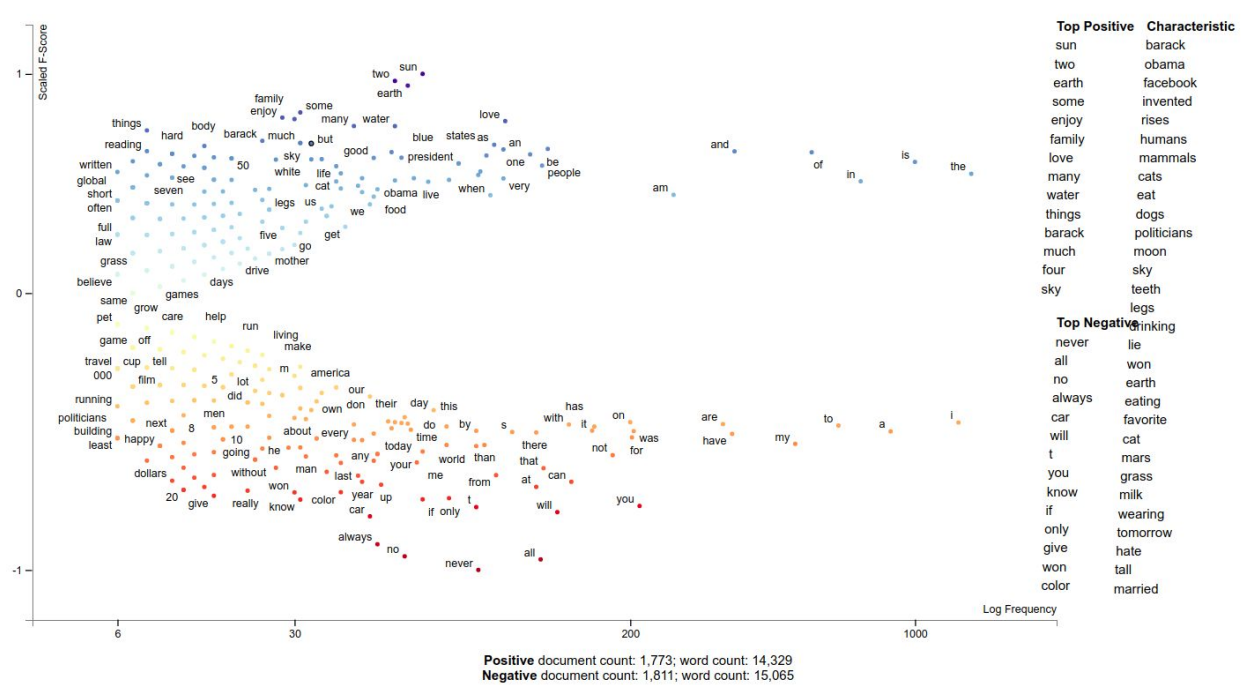
Scenario/ Metric	**Baseline**	**Best Feature Set without vocabulary augmentation**	**Best Feature Set with vocabulary augmentation**
Precision	0.5617701290719115	0.5606936416184971	0.5633314054366686
Recall	0.544047619047619	0.5773809523809523	0.5797619047619048
F1 score	0.552766858179619	0.5689149560117301	0.5714285714285715
Accuracy	0.5611275964391691	0.5637982195845698	0.5664688427299703

Linguistic Inquiry and Word Count (LIWC)

- LIWC variables are percentages of total word within a text

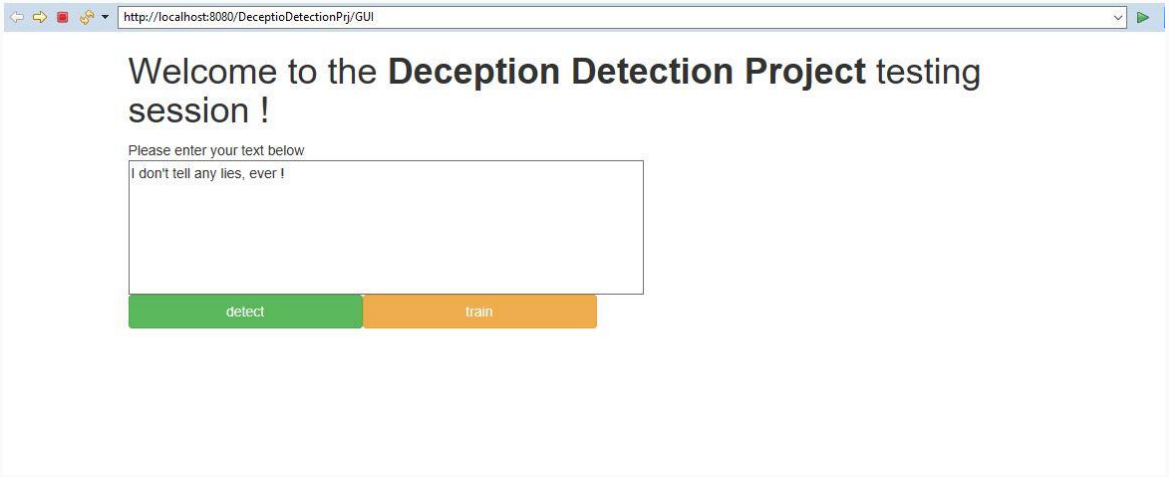
Variables	**Lie**	**Truth**
Emotional Tone	50.70	99.00
Function Words (Negation)	1.59	0.89
Affect (Positive Emotion)	2.56	12.80
Affect (Negative Emotion)	1.22	1.07
Social	5.81	5.33
Certainty (Certain)	1.77	10.75

Scattertext Tool (Python Pre/Post-processing)



GUI Demo

https://deceptiondetection.herokuapp.com/GUI



Conclusion and Future Work

- Increase data quality
- Improve vocabulary augmentation (i.e. word2vec)

References

- Veronica Perez-Rosas and Rada Mihalcea, Experiments in Open Domain Deception Detection, in Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP 2015), Lisbon, Portugal, September 2015, http://web.eecs.umich.edu/~mihalcea/papers/perezrosas.emnlp15.pdf
- Open Domain Deception Detection Dataset https://web.eecs.umich.edu/~mihalcea/downloads.html#OpenDeception
- Linguistic Inquiry and Word Count http://liwc.wpengine.com/
- Wordnet Database Files (A lexical database of English which records semantic relationships between words) https://wordnet.princeton.edu
- Stanford Core NLP Parser https://nlp.stanford.edu/software/lex-parser.shtml

Thank You!