Detecting Clickbait: This Poster Will Leave You Speechless

Iryna Zayats
izayats@stanford.edu

MOTIVATION

In competition for reader's attention, online outlets have produced a phenomena known as clickbait — **headlines engineered to exploit human psychology** and entice the reader to follow a link, often to an article of dubious value.

We have all seen these headlines:

- *Only People with IQ Over 160 Can Pass This Test*
- *7 Signs Your Partner Is Cheating (You Would Never Guess Number 5)*
- *Man Hugs a Wild Lion. You Won't Believe What Happens Next!*

Clickbait has become more than just a minor nuisance to Internet users. Aside from adding to mental fatigue and general reader dissatisfaction, it has become a vehicle for social bots, fake news, and spread of malicious software.

Project goal: use machine learning to detect clickbait-y headlines. Specifically, compare two approaches to detecting clickbait:

- one built on expert knowledge and manual features
- another based solely on word embeddings

DATA

- Dataset for the project has been taken from Clickbait Challenge[1], a 2017 competition by Bauhaus-Universit¨at Weimar
- Includes **22,033 articles and their headlines** shared by different news outlets via Twitter
- Each headline has been **evaluated for "clickbaitness" by at least five human evaluators** on a 4-point scale:

`not clickbaiting` `slightly clickbaiting` `considerably clickbaiting` `heavy clickbaiting`

- The scale was then converted to [0, 0.33, 0.66, 1], and a mean value of 5 evaluations considered as *clickbait score* of a given headline.
- The model's task is to predict this clickbait score (a value between 0 and 1) with high accuracy.

CHALLENGES

- Clickbait a very subjective measure. Two people looking at the same headline may strongly disagree whether it's clickbait or not.
- MSE (Mean Squared Error) between each annotator's judgements and the mean judgement for the same article is quite high, 0.072.
- High variance in how true score is assigned may make it difficult to fit the model to data.

APPROACH: MANUAL FEATURES

From studying the dataset, clickbait titles tend to:

- begin with a number *("15 surprising facts about Tesla cars you probably didn't know")*
- include question or exclamation marks or other unusual punctuation *("...And now the circle is complete.")*
- use determiner words like this, his, their, your, which, what *("This new cookie is causing some seriously weird side effects")*
- contain little factual information *("Impressive. Most impressive.")*
- address the reader rather than third parties of facts *("This Satisfying Food Quiz Will Reveal Where You Should Live")*
- contain sensational, hyperbolic words, and superlatives

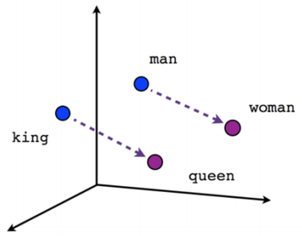
Features
Number of words
Contains a question
Contains an exclamation
Starts with a number
Sentiment analysis: polarity
Sentiment analysis: subjectivity
Ratio of determiner words
Number of superlatives
Number of stop words
Number of nouns
Number of adjectives
Number of verbs

Regression model:

- Models evaluated: Ridge, XGB, SVR, DecisionTree, RandomForest, GradientBoosting, CatBoost.
- Hyper-parameters tuned via grid search, 75/25% train/test data split.
- **Best performing model**: CatBoostRegressor

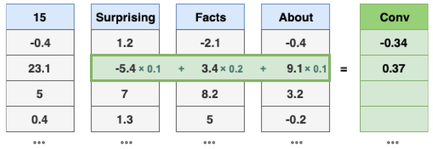
APPROACH: WORD EMBEDDINGS

- What if our intuition about what makes a clickbait-y headline is off, difficult to formalize or convert to features?
- Can we somehow use raw words in a regression model? ML methods work best with numerical/categorical features, so we need a way to convert words to numbers in a meanigful way.
- Solution: **represent words as points in a high-dimentional space based on co-occurrence and context** they are used in (a large external corpus of text is used to pre-train these embeddings, for this project: GloVe[2]).
- Numerical representation is based on context and semantics, so the resulting model is meanigful.
- Similar words have high cosine similarity, and addition/subtraction carries semantic meaning → this works well with convolutional filters

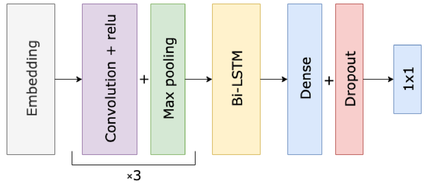


`king` - `man` + `woman` ≈ `queen`

Convolutions on word embeddings:



Best performing architecture: CNN + Bi-LSTM:



RESULTS

Word embeddings + DL outperformed the manual features model:

	MSE	Precision	Recall	Accuracy	F1
Regression with manual features	0.039	79.8%	70.6%	70.6%	0.73
Deep learning with word embeddings	0.037	77.8%	75.0%	83.5%	0.76

If this model were a part of 2017 Clickbait Challenge, it would have placed 6th out of 16 contestants.

In the real world

I ran the best performing classifier against the Upcoming section of Hacker News[3], a news aggregator known for its anti-clickbait stance. Here are some headlines classified as clickbait-y:

- *15 billionaires and 6 corporations owning the majority of U.S media outlets*
- *Your Debugger Sucks*
- *A brutally honest landing page*

At least 2 of the headlines classified as clickbait-y have made it to the front page of Hacker News, at which point they have been edited (most likely, by site's moderators), according to Hacker News Title Edit Tracker[4]:

`No more Google for console junkies` ⤵

`I am no longer able to use Google with Lynx`

`You Don't Bring Bad News to the Cult Leader": Inside the Fall of WeWork` ⤵

`The Fall of WeWork`

This suggests that the classifier could be a useful tool in bringing a moderator's attention to potentially clickbait-y headlines.

References

[1] Clickbait Challenge, https://www.clickbait-challenge.org/

[2] GloVe: Global Vectors for Word Representation, https://nlp.stanford.edu/projects/glove/

[3] Hacker News, https://news.ycombinator.com/

[4] Hacker News Title Edit Tracker, https://hackernewstitles.netlify.com/

Video presentation: https://cutt.ly/clickbait