The Language Modeling problem

Setup

- Assume a (finite) vocabulary of words:
$$\mathcal{V} = \{\textit{killer}, \textit{crazy}, \textit{clown}\}$$
- Use $\mathcal{V}$ to construct an infinite set of *sentences*
$$\begin{aligned}\mathcal{V}^+ = \ \{ & \\ & \text{clown, killer clown, crazy clown,} \\ & \text{crazy killer clown, killer crazy clown,} \\ & \ldots \\ \} & \end{aligned}$$
- A *sentence* is **defined** as each $s \in \mathcal{V}^+$

The Language Modeling problem

Data

Given a training data set of example sentences $s \in \mathcal{V}^+$

Language Modeling problem

Estimate a probability model:

$$\sum_{s \in \mathcal{V}^+} p(s) = 1.0$$

- p(clown) = 1e-5
- p(killer) = 1e-6
- p(killer clown) = 1e-12
- p(crazy killer clown) = 1e-21
- p(crazy killer clown killer) = 1e-110
- p(crazy clown killer killer) = 1e-127

Why do we want to do this?

Scoring Hypotheses in Speech Recognition

From acoustic signal to candidate transcriptions

Hypothesis	Score
the station signs are in deep in english	-14732
the stations signs are in deep in english	-14735
the station signs are in deep into english	-14739
the station 's signs are in deep in english	-14740
the station signs are in deep in the english	-14741
the station signs are indeed in english	-14757
the station 's signs are indeed in english	-14760
the station signs are indians in english	-14790
the station signs are indian in english	-14799
the stations signs are indians in english	-14807
the stations signs are indians and english	-14815

Scoring Hypotheses in Machine Translation

From source language to target language candidates

Hypothesis	Score
we must also discuss a vision .	-29.63
we must also discuss on a vision .	-31.58
it is also discuss a vision .	-31.96
we must discuss on greater vision .	-36.09
⋮	⋮

Scoring Hypotheses in Decryption

Character substitutions on ciphertext to plaintext candidates

Hypothesis	Score
Heopaj, zk ukq swjp pk gjks w oaynap?	-93
Urbcnw, mx hxd fjwc cx twxf j bnlanc?	-92
Wtdepy, oz jzf hlye ez vyzh l dpncpe?	-91
Mjtufo, ep zpv xbou up lopx b tfdsfu?	-89
Nkuvgp, fq aqw ycpv vq mpqy c ugetgv?	-87
Gdnozi, yj tjp rvio oj fijr v nzxmzo?	-86
Czjkve, uf pfl nrek kf befn r jvtivk?	-85
Yvfgra, qb lbh jnag gb xabj n frperg?	-84
Zwghsb, rc mci kobh hc ybck o gsqfsh?	-83
Byijud, te oek mqdj je adem q iushuj?	-77
Jgqrcl, bm wms uylr rm ilmu y qcapcr?	-76
Listen, do you want to know a secret?	-25

Scoring Hypotheses in Spelling Correction

Substitute spelling variants to generate hypotheses

Hypothesis	Score
… stellar and versatile **acress** whose combination of sass and glamour has defined her …	-18920
… stellar and versatile **acres** whose combination of sass and glamour has defined her …	-10209
… stellar and versatile **actress** whose combination of sass and glamour has defined her …	-9801

T9 to English

Grover, King, & Kushler. 1998.
Reduced keyboard disambiguating computer. US Patent 5,818,437

Sequence of numbers to English

Input	Hypothesis	Score
46 04663	GO HOOD	-24
46 04663	GO HOME	-10
843 0746453	?	?
06678 07678527		
0243373 0460843		
096753		

Probability models of language

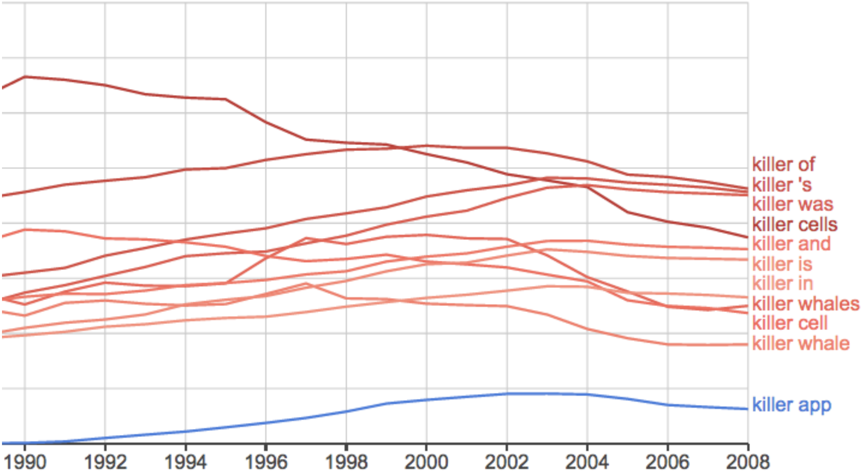
Question

- Given a finite vocabulary set $\mathcal{V}$
- We want to build a probability model $P(s)$ for all $s \in \mathcal{V}^+$
- **But** we want to consider sentences s of each length ℓ separately.
- Write down a new model over $\mathcal{V}^+$ such that $P(s \mid \ell)$ is in the model
- **And** the model should be equal to $\sum_{s \in \mathcal{V}^+} P(s)$.
- Write down the model

$$\sum_{s \in \mathcal{V}^+} P(s) = \ldots$$

n-gram Models

Google *n*-gram viewer



Number of Parameters

How many probabilities in each n-gram model

- Assume $\mathcal{V} = \{killer, crazy, clown, UNK\}$

Question

How many unigram probabilities: $P(x)$ for $x \in \mathcal{V}$?

4

Number of Parameters

How many probabilities in each *n*-gram model

- Assume $\mathcal{V} = \{\textit{killer}, \textit{crazy}, \textit{clown}, \textit{UNK}\}$

Question

How many bigram probabilities: $P(y|x)$ for $x, y \in \mathcal{V}$?

$4^2 = 16$

Number of Parameters

How many probabilities in each *n*-gram model

- Assume $\mathcal{V} = \{killer, crazy, clown, UNK\}$

Question

How many trigram probabilities: $P(z|x, y)$ for $x, y, z \in \mathcal{V}$?

$4^3 = 64$

Number of Parameters

Question

- Assume $|\mathcal{V}| = 50{,}000$ (a realistic vocabulary size for English)
- What is the minimum size of training data in tokens?
  - If you wanted to observe all unigrams at least once.
  - If you wanted to observe all trigrams at least once.

125,000,000,000,000 (125 Ttokens)

Some trigrams should be zero since they do not occur in the language, $P(\mathit{the} \mid \mathit{the}, \mathit{the})$.
But others are simply unobserved in the training data, $P(\mathit{idea} \mid \mathit{colourless}, \mathit{green})$.

Evaluating Language Models

- So far we've seen the probability of a sentence: $P(w_0, \ldots, w_n)$
- What is the probability of a collection of sentences, that is what is the probability of an unseen test corpus T
- Let $T = s_0, \ldots, s_m$ be a test corpus with sentences s_i
- T is assumed to be separate from the training data used to train our language model $P(s)$
- What is $P(T)$?

Evaluating Language Models: Independence assumption

- $T = s_0, \ldots, s_m$ is the text corpus with sentences s_0 through s_m
- $P(T) = P(s_0, s_1, s_2, \ldots, s_m)$ – but each sentence is independent from the other sentences
- $P(T) = P(s_0) \cdot P(s_1) \cdot P(s_2) \cdot \ldots \cdot P(s_m) = \prod_{i=0}^{m} P(s_i)$
- $P(s_i) = P(w_0^{(i)}, \ldots, w_{n_i}^{(i)})$ – which can be any n-gram language model
- A language model is better if the value of $P(T)$ is higher for unseen sentences T, we want to maximize:

$$P(T) = \prod_{i=0}^{m} P(s_i)$$

Evaluating Language Models: Computing the Average

- However, T can be any arbitrary size
- $P(T)$ will be lower if T is larger.
- Instead of the probability for a given T we can compute the *average* probability.
- M is the total number of tokens in the test corpus T:

$$M = \sum_{i=0}^{m} \text{length}(s_i)$$

- The average *log* probability of the test corpus T is:

$$\frac{1}{M} \log_2 \prod_{i=0}^{m} P(s_i) = \frac{1}{M} \sum_{i=0}^{m} \log_2 P(s_i)$$

Evaluating Language Models: Perplexity

- The average *log* probability of the test corpus T is:

$$\ell = \frac{1}{M} \sum_{i=0}^{m} \log_2 P(s_i)$$

- Note that ℓ is a negative number
- We evaluate a language model using *Perplexity* which is $2^{-\ell}$

Evaluating Language Models

Question

Show that:

$$2^{-\frac{1}{M}\log_2 \prod_{i=0}^{m} P(s_i)} = \frac{1}{\sqrt[M]{\prod_{i=0}^{m} P(s_i)}}$$

Acknowledgements

Many slides borrowed or inspired from lecture notes by Michael Collins, Chris Dyer, Kevin Knight, Chris Manning, Philipp Koehn, Adam Lopez, Graham Neubig, Richard Socher and Luke Zettlemoyer from their NLP course materials.

All mistakes are my own.

A big thank you to all the students who read through these notes and helped me improve them.