Language Modeling with Power Low Rank Ensembles

Ankur Parikh
Avneesh Saluja
Chris Dyer
Eric Xing
Overview
Overview

• **Model:** Framework for language modeling using ensembles of low rank matrices and tensors

• **Relations:** Includes existing $n$-gram smoothing techniques as special cases
Overview

- **Model**: Framework for language modeling using ensembles of low rank matrices and tensors

- **Relations**: Includes existing $n$-gram smoothing techniques as special cases

- **Performance**: Consistently outperforms state-of-the-art Kneser Ney baselines for same context length

- **Speed**: Easily scalable since no partition function required
Outline

• Introduction

• Background on $n$-gram smoothing

• Our Approach
  • Rank
  • Power
  • Constructing the Ensemble

• Experiments
Language Modeling

• Evaluate probabilities of sentences
Language Modeling

• Evaluate probabilities of sentences

Linear algebra is awesome
Language Modeling

• Evaluate probabilities of sentences

Linear algebra is awesome

\[ P(w_1, \ldots, w_4) = 0.3648 \]
Language Modeling

• Evaluate probabilities of sentences

\[ P(w_1, \ldots, w_4) = 0.3648 \]

Linear algebra is awesome

Linear algebra is boring
Language Modeling

• Evaluate probabilities of sentences

\[ P(w_1, \ldots, w_4) = 0.3648 \]

\[ P(w_1, \ldots, w_4) = 0.1922 \]

*Linear algebra is awesome*

*Linear algebra is boring*
Language Modeling

• Evaluate probabilities of sentences

\[ P(w_1, \ldots, w_4) = 0.3648 \]
\[ P(w_1, \ldots, w_4) = 0.1922 \]

• Very useful in downstream applications such as machine translation and speech recognition.
N-grams

- Predominant approach to language modeling
N-grams

- Predominant approach to language modeling
$N$-grams

- Predominant approach to language modeling

$w_i$
N-grams

- Predominant approach to language modeling

\[ \text{count}(w_i) \]
\textbf{N-grams}

- Predominant approach to language modeling

\[ \text{count}(w_i) \]
N-grams

• Predominant approach to language modeling

\[ \text{count}(w_i) \]
**N-grams**

- Predominant approach to language modeling

\[
\text{count}(w_i, w_{i-1}) \quad \text{count}(w_i)
\]
N-grams

- Predominant approach to language modeling

\[ \text{count}(w_i, w_{i-1}) \]

\[ \text{count}(w_i) \]
\textbf{N-grams}

- Predominant approach to language modeling

\[ \text{count}(w_i, w_{i-1}) \]

\[ \text{count}(w_i) \]
**N-grams**

- Predominant approach to language modeling

$$\text{count}(w_i, w_{i-1}, w_{i-2})$$

$$\text{count}(w_i, w_{i-1})$$

$$\text{count}(w_i)$$
\( \hat{P}(w_i|w_{i-1}, w_{i-2}) \)  
\( \hat{P}(w_i|w_{i-1}) \)  
\( \hat{P}(w_i) \)

- Alleviate data sparsity problem

\( P(w_i) = \frac{P(w_i|w_{i-1}) \cdot P(w_{i-1})}{\sum_{w_{i-1}} P(w_i|w_{i-1}) \cdot P(w_{i-1})} \)
$N$-gram Smoothing

• Alleviate data sparsity problem

\[
\hat{P}(w_i|w_{i-1}, w_{i-2}) \quad \hat{P}(w_i|w_{i-1}) \quad \hat{P}(w_i)
\]
\textbf{N-gram Smoothing}

- Alleviate data sparsity problem

\begin{align*}
\hat{P}(w_i|w_{i-1}, w_{i-2}) \\
\hat{P}(w_i|w_{i-1}) \\
\hat{P}(w_i)
\end{align*}
N-gram Smoothing

• Alleviate data sparsity problem

\[
\hat{P}(w_i|w_{i-1}, w_{i-2}) \quad \hat{P}(w_i|w_{i-1}) \quad \hat{P}(w_i)
\]
\( \hat{P}(w_i|w_{i-1}, w_{i-2}) \)  
\( \hat{P}(w_i|w_{i-1}) \)  
\( \hat{P}(w_i) \)

- Alleviate data sparsity problem

\[ \hat{P}(w_i|w_{i-1}, w_{i-2}) = \frac{\hat{P}(w_i|w_{i-1}) \cdot \hat{P}(w_{i-2}|w_{i-1})}{\hat{P}(w_i)} \]
N-gram Smoothing

• Alleviate data sparsity problem

\[ \hat{P}(w_i|w_{i-1}, w_{i-2}) \]

\[ \hat{P}(w_i|w_{i-1}) \]

\[ \hat{P}(w_i) \]
Advantages of \( N \)-gram Models

- “Fine-to-coarse”, captures various levels of dependence

\[
\hat{P}(w_i|w_{i-1}, w_{i-2}) \quad \hat{P}(w_i|w_{i-1}) \quad \hat{P}(w_i)
\]

- Very fast
  - \( O(N) \) test complexity
  - Low context sizes sufficient
Classic Disadvantage of $N$-gram Models

- No notion of similarity between words

\[
\hat{P}(w_i|w_{i-1}) \quad \text{and} \quad \hat{P}(w_i)
\]
Classic Disadvantage of $N$-gram Models

- No notion of similarity between words

\[
\hat{P}(w_i|w_{i-1})
\]

\[
\hat{P}(w_i)
\]

(house, decrepit)
Classic Disadvantage of $N$-gram Models

- No notion of similarity between words

$$\hat{P}(w_i | w_{i-1})$$

(house, decrepit)

$$\hat{P}(w_i)$$

(house)
Classic Disadvantage of $N$-gram Models

- No notion of similarity between words

$P(w_i | w_{i-1})$

$P(w_i)$

(house, decrepit)

(house, old)

(house, shabby)
Classic Disadvantage of $N$-gram Models

• No notion of similarity between words

$$
\hat{P}(w_i | w_{i-1})
$$

$\hat{P}(w_i)$

(house, decrepit)  
(house, old)  
(house, shabby)  
(house)
Classic Disadvantage of $N$-gram Models

- No notion of similarity between words

\[
\hat{P}(w_i|w_{i-1}) \quad \hat{P}(w_i)
\]

(house, decrepit) \hspace{2cm} (house, old) \hspace{2cm} (house, {synonym of old})

(house, shabby)
Classic Disadvantage of $N$-gram Models

- No notion of similarity between words

$$\hat{P}(w_i|w_{i-1})$$

- (house, decrepit)
- (house, old)
- (house, shabby)
- (house, {synonym of old})

$$\hat{P}(w_i)$$

- (house)
Motivation For Low Rank Methods

• Project words to lower-dimensional space
Motivation For Low Rank Methods

• Project words to lower-dimensional space

\[ \approx \]
Motivation For Low Rank Methods

- Project words to lower-dimensional space

- Words with similar contexts will have similar projections
Motivation For Low Rank Methods

• Project words to lower-dimensional space

[Diagram showing projection of words to lower-dimensional space]

• Words with similar contexts will have similar projections

- house
- cabin
- flat
Motivation For Low Rank Methods

• Project words to lower-dimensional space

Words with similar contexts will have similar projections

- House
- Cabin
- Flat
- Old
- Shabby
- Decrepit
Low Rank Approaches
Low Rank Approaches

• Low rank approximation successful in many ML applications
  • Collaborate filtering (Netflix)
  • Matrix completion
Low Rank Approaches

- Low rank approximation successful in many ML applications
  - Collaborate filtering (Netflix)
  - Matrix completion

- These solutions have been attempted in language modeling
  - Saul and Pereira 1997
  - Hutchinson et al. 2011
Low Rank Approaches

• Low rank approximation successful in many ML applications
  • Collaborate filtering (Netflix)
  • Matrix completion

• These solutions have been attempted in language modeling
  • Saul and Pereira 1997
  • Hutchinson et al. 2011

• Unfortunately, not generally competitive with Kneser Ney
Problem: Low Rank Methods
Operate at Fixed Granularity

If rank is too small......
Problem: Low Rank Methods Operate at Fixed Granularity

If rank is too small......

(break, spring)
Problem: Low Rank Methods Operate at Fixed Granularity

If rank is too small......

\[ \approx \]

Probability gets diluted since “break” has many synonyms
Problem: Low Rank Methods Operate at Fixed Granularity

If rank is too large....
Problem: Low Rank Methods Operate at Fixed Granularity

If rank is too large....

(domicile, dilapidated)
Problem: Low Rank Methods Operate at Fixed Granularity

If rank is too large....

Probabilities of rare words a problem, since representation is too fine grained

(domicile, dilapidated)
Our Approach
Our Approach

• Construct ensembles of low rank matrices/tensors to model language at multiple granularities
Our Approach

• Construct ensembles of low rank matrices/tensors to model language at multiple granularities

• Includes existing $n$-gram techniques as special cases
  • Absolute discounting
  • Jelinek Mercer (deleted-interpolation)
  • Kneser Ney
Our Approach

• Construct ensembles of low rank matrices/tensors to model language at multiple granularities

• Includes existing $n$-gram techniques as special cases
  • Absolute discounting
  • Jelinek Mercer (deleted-interpolation)
  • Kneser Ney

• Preserves advantages of standard $n$-gram approaches
  • Effective for short context lengths
  • Fast evaluation at test time
Outline

• Introduction

• Background on *Kneser Ney* smoothing

• Our Approach
  • Rank
  • Power
  • Constructing the Ensemble

• Experiments
Kneser Ney - Intuition

• Lower order distribution should be altered
Kneser Ney - Intuition

• Lower order distribution should be altered

• Consider two words, York and door
  • York only follows very few words i.e. New York
  • Door can follow many words i.e. “the door”, “red door”, “my door” etc.

\[ P(w_i = \text{door} \mid \text{backed} - \text{off on } w_{i-1}) > P(w_i = \text{York} \mid \text{backed} - \text{off on } w_{i-1}) \]
**Kneser Ney - Intuition**

- Lower order distribution should be altered
  
- Consider two words, *York* and *door*
  
  - *York* only follows very few words i.e. New York
  
  - *Door* can follow many words i.e. “the door”, “red door”, “my door” etc.

\[
P(w_i = \text{door} \mid \text{backed} \rightarrow \text{off on } w_{i-1}) > P(w_i = \text{York} \mid \text{backed} \rightarrow \text{off on } w_{i-1})
\]
Kneser Ney Unigram Distribution

\[ N_-(w_i) = |\{w : c(w_i, w) > 0\}| \]

Diversity of \( w_i \)'s history
Kneser Ney Unigram Distribution

\[ N_-(w_i) = |\{w : c(w_i, w) > 0\}| \]

Diversity of \( w_i \)'s history

\[ \hat{P}_{kn-uni}(w_i) = \frac{N_-(w_i)}{\sum_w N_-(w)} \]
Discounting

\[ \hat{P}_d(w_i | w_{i-1}) = \frac{\max(c(w_i, w_{i-1}) - d, 0)}{\sum_w c(w, w_{i-1})} \]
Discounting

\[ \hat{P}_d(w_i | w_{i-1}) = \frac{\max(c(w_i, w_{i-1}) - d, 0)}{\sum_w c(w, w_{i-1})} \]

\[ \hat{P}_{kney}(w_i | w_{i-1}) = \hat{P}_d(w_i | w_{i-1}) + \gamma(w_{i-1}) \hat{P}_{kn-uni}(w_i) \]
Discounting

\[ \hat{P}_d(w_i | w_{i-1}) = \frac{\max(c(w_i, w_{i-1}) - d, 0)}{\sum_w c(w, w_{i-1})} \]

\[ \hat{P}_{kney}(w_i | w_{i-1}) = \hat{P}_d(w_i | w_{i-1}) + \gamma(w_{i-1})\hat{P}_{kn-uni}(w_i) \]

Where \( \gamma(w_{i-1}) \) is the leftover probability
Lower Order Marginal Aligns!

\[
\hat{P}(w_i) = \sum_{w_{i-1}} \hat{P}_{kney}(w_i | w_{i-1}) \hat{P}(w_{i-1})
\]
Generalizing KN to PLRE

Kneser Ney

Power Low Rank Ensembles
Generalizing KN to PLRE

**Kneser Ney**

- Ensemble composed of unsmoothed $n$-grams

**Power Low Rank Ensembles**
Generalizing KN to PLRE

**Kneser Ney**

- Ensemble composed of unsmoothed $n$-grams

- Alter lower order distributions by using count of unique histories

**Power Low Rank Ensembles**
Generalizing KN to PLRE

**Kneser Ney**

- Ensemble composed of unsmoothed $n$-grams

- Alter lower order distributions by using count of unique histories

- Use absolute discounting to interpolate different $n$-grams and preserve lower order marginal constraint

**Power Low Rank Ensembles**
# Generalizing KN to PLRE

<table>
<thead>
<tr>
<th>Kneser Ney</th>
<th>Power Low Rank Ensembles</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Ensemble composed of unsmoothed $n$-grams</td>
<td>?</td>
</tr>
<tr>
<td></td>
<td>?</td>
</tr>
<tr>
<td>- Alter lower order distributions by using count of unique histories</td>
<td>?</td>
</tr>
<tr>
<td></td>
<td>?</td>
</tr>
<tr>
<td>- Use absolute discounting to interpolate different $n$-grams and</td>
<td>?</td>
</tr>
<tr>
<td>preserve lower order marginal constraint</td>
<td>?</td>
</tr>
</tbody>
</table>
Generalizing KN to PLRE

**Kneser Ney**

- Ensemble composed of unsmoothed $n$-grams

- Alter lower order distributions by using count of unique histories

- Use absolute discounting to interpolate different $n$-grams and preserve lower order marginal constraint

**Power Low Rank Ensembles**

- ?

- ?

- ?
In General, Bigram is Full Rank
Independence = Rank 1

- If $w_i$ and $w_{i-1}$ are independent

$$P(w_i, w_{i-1}) = P(w_i)P(w_{i-1})$$
Independence = Rank 1

• If $w_i$ and $w_{i-1}$ are independent

$$P(w_i, w_{i-1}) = P(w_i)P(w_{i-1})$$
Independence = Rank 1

• If \( w_i \) and \( w_{i-1} \) are independent

\[
P(w_i, w_{i-1}) = P(w_i)P(w_{i-1})
\]
Independence = Rank 1

• If \( w_i \) and \( w_{i-1} \) are independent

\[
P(w_i, w_{i-1}) = P(w_i)P(w_{i-1})
\]

• But what if \( w_i \) and \( w_{i-1} \) are not independent? What does the best rank 1 approximation give?
• Let $B$ be the matrix such that
$$B(w_i, w_{i-1}) = c(w_i, w_{i-1})$$

• Let
$$M_1 = \min_{M: \text{rank}(M)=1} \|B - M\|_{KL}$$

• Then
$$M_1(w_i, w_{i-1}) \propto \hat{P}(w_i)\hat{P}(w_{i-1})$$

Generalized KL

[Lee and Seung 2001]
Rank

• MLE unigram is normalized rank 1 approx. of MLE bigram under KL:

\[
\hat{P}(w_i) = \frac{M_1(w_i, w_{i-1})}{\sum_{w_i} M_1(w_i, w_{i-1})}
\]
• MLE unigram is normalized rank 1 approx. of MLE bigram under KL:

\[
\hat{P}(w_i) = \frac{M_1(w_i, w_{i-1})}{\sum_{w_i} M_1(w_i, w_{i-1})}
\]

• Vary rank to obtain quantities between bigram and unigram
• MLE unigram is normalized rank 1 approx. of MLE bigram under KL:

\[
\hat{P}(w_i) = \frac{M_1(w_i, w_{i-1})}{\sum_{w_i} M_1(w_i, w_{i-1})}
\]

• Vary rank to obtain quantities between bigram and unigram
• MLE unigram is normalized rank 1 approx. of MLE bigram under KL:

\[ \hat{P}(w_i) = \frac{M_1(w_i, w_{i-1})}{\sum_w M_1(w, w_{i-1})} \]

• Vary rank to obtain quantities between bigram and unigram
Generalizing KN to PLRE

**Kneser Ney**

- Ensemble composed of unsmoothed $n$-grams
- Alter lower order distributions by using count of unique histories
- Use absolute discounting to interpolate different $n$-grams and preserve lower order marginal constraint

**Power Low Rank Ensembles**

- Ensemble composed of unsmoothed $n$-grams plus other low rank matrices/tensors
Generalizing KN to PLRE

**Kneser Ney**

- Ensemble composed of unsmoothed $n$-grams

- Alter lower order distributions by using count of unique histories

- Use absolute discounting to interpolate different $n$-grams and preserve lower order marginal constraint

**Power Low Rank Ensembles**

- Ensemble composed of unsmoothed $n$-grams plus other low rank matrices/tensors
Consider Elementwise Power
Consider Elementwise Power

\[
B = \begin{bmatrix}
1 & 2 & 1 \\
0 & 5 & 0 \\
2 & 0 & 0 \\
\end{bmatrix}
\]
Consider Elementwise Power

\[ B = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 5 & 0 \\ 2 & 0 & 0 \end{bmatrix} \]

Row sum:

\[ \begin{bmatrix} 4 \\ 5 \\ 2 \end{bmatrix} \]
Consider Elementwise Power

\[ B = \begin{bmatrix}
1 & 2 & 1 \\
0 & 5 & 0 \\
2 & 0 & 0
\end{bmatrix} \]

Row sum:

\[ \begin{bmatrix}
4 \\
5 \\
2
\end{bmatrix} \]
Consider Elementwise Power

\[
\begin{bmatrix}
1 & 2 & 1 \\
0 & 5 & 0 \\
2 & 0 & 0 \\
\end{bmatrix}
\rightarrow
\begin{bmatrix}
4 \\
5 \\
2 \\
\end{bmatrix}
\rightarrow
\begin{bmatrix}
1 & 1.4 & 1 \\
0 & 2.2 & 0 \\
1.4 & 0 & 0 \\
\end{bmatrix}
\]
Consider Elementwise Power

\[
\begin{bmatrix}
1 & 2 & 1 \\
0 & 5 & 0 \\
2 & 0 & 0
\end{bmatrix}
\rightarrow
\begin{bmatrix}
1 & 1.4 & 1 \\
0 & 2.2 & 0 \\
1.4 & 0 & 0
\end{bmatrix}
\]

\[
\begin{bmatrix}
4 \\
5 \\
2
\end{bmatrix}
\rightarrow
\begin{bmatrix}
3.4 \\
2.2 \\
1.4
\end{bmatrix}
\]
Consider Elementwise Power

$$B = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 5 & 0 \\ 2 & 0 & 0 \end{bmatrix}$$

$$B^{0.5} = \begin{bmatrix} 1 & 1.4 & 1 \\ 0 & 2.2 & 0 \\ 1.4 & 0 & 0 \end{bmatrix}$$

Row sum of $B$:
$$\begin{bmatrix} 4 \\ 5 \\ 2 \end{bmatrix}$$

Row sum of $B^{0.5}$:
$$\begin{bmatrix} 3.4 \\ 2.2 \\ 1.4 \end{bmatrix}$$
Consider Elementwise Power

\[
B = \begin{bmatrix}
1 & 2 & 1 \\
0 & 5 & 0 \\
2 & 0 & 0 \\
\end{bmatrix} \quad B^{0.5} = \begin{bmatrix}
1 & 1.4 & 1 \\
0 & 2.2 & 0 \\
1.4 & 0 & 0 \\
\end{bmatrix} \quad B^{0} = \begin{bmatrix}
1 & 1 & 1 \\
0 & 1 & 0 \\
1 & 0 & 0 \\
\end{bmatrix}
\]

Row sums:

- Row sum of \(B\): \(4, 5, 2\)
- Row sum of \(B^{0.5}\): \(3.4, 2.2, 1.4\)
- Row sum of \(B^{0}\): \(1, 1, 1\)
Consider Elementwise Power

\[
B = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 5 & 0 \\ 2 & 0 & 0 \end{bmatrix} \quad \rightarrow \quad B^{0.5} = \begin{bmatrix} 1 & 1.4 & 1 \\ 1.4 & 2.2 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad \rightarrow \quad B^0 = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}
\]

Row sum:
- \(B\) has row sums \(4, 5, 2\)
- \(B^{0.5}\) has row sums \(3.4, 2.2, 1.4\)
- \(B^0\) has row sums \(3, 1, 1\)
Consider Elementwise Power

<table>
<thead>
<tr>
<th>B</th>
<th>$B^{0.5}$</th>
<th>$B^0$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\begin{bmatrix} 1 &amp; 2 &amp; 1 \ 0 &amp; 5 &amp; 0 \ 2 &amp; 0 &amp; 0 \end{bmatrix}$</td>
<td>$\begin{bmatrix} 1 &amp; 1.4 &amp; 1 \ 0 &amp; 2.2 &amp; 0 \ 1.4 &amp; 0 &amp; 0 \end{bmatrix}$</td>
<td>$\begin{bmatrix} 1 &amp; 1 &amp; 1 \ 0 &amp; 1 &amp; 0 \ 1 &amp; 0 &amp; 0 \end{bmatrix}$</td>
</tr>
</tbody>
</table>

emphasis on diversity
Consider Elementwise Power
Consider Elementwise Power

\[ M_1^0 = \min_{M : M \geq 0, \text{rank}(M) = 1} \| B^0 - M \|_{KL} \]
Consider Elementwise Power

\[ \begin{align*}
M_1^0 &= \min_{M : M \geq 0, \text{rank}(M) = 1} \| B^0 - M \|_{KL} \\
\hat{P}_{kn-un}(w_i) &= \frac{M_1^0(w_i, w_{i-1})}{\sum_w M_1^0(w, w_{i-1})}
\end{align*} \]
Consider Elementwise Power

\[ M_1^0 = \min_{M : M \geq 0, \text{rank}(M) = 1} \| B^0 - M \|_{KL} \]

\[ \hat{P}_{\text{kn-unif}}(w_i) = \frac{M_1^0(w_i, w_{i-1})}{\sum_w M_1^0(w, w_{i-1})} \]

power = 1
full rank
Consider Elementwise Power

\[ M_1^0 = \min_{M: M \geq 0, \text{rank}(M)=1} \| B^0 - M \|_{KL} \]

\[ \hat{P}_{\text{kn-un}}(w_i) = \frac{M_1^0(w_i, w_{i-1})}{\sum_w M_1^0(w, w_{i-1})} \]

- power = 1, full rank
- power = 0, full rank

\[ \begin{array}{c}
\text{power} \\
\text{full rank}
\end{array} \rightarrow \begin{array}{c}
\text{power} \\
\text{full rank}
\end{array} \]
Consider Elementwise Power

\[ M_1^0 = \min_{M: M \geq 0, \text{rank}(M) = 1} \| B^0 - M \|_{KL} \]

\[ \hat{P}_{kn-uni}(w_i) = \frac{M_1^0(w_i, w_i-1)}{\sum_w M_1^0(w, w_i-1)} \]

- power = 1 full rank
- power = 0 full rank
- power = 0 rank = 1
Varying Rank and Power

• Construct matrices of varying rank and power

power = 1
full rank

power = 0
rank = 1
Varying Rank and Power

• Construct matrices of varying rank and power

- power = 1 (full rank)
- power = 0.5 (low rank)
- power = 0 (rank = 1)
Varying Rank and Power

- Generalizes to higher orders
**Generalizing KN to PLRE**

**Kneser Ney**
- Ensemble composed of unsmoothed $n$-grams
- Alter lower order distributions by using count of unique histories
- Use absolute discounting to interpolate different $n$-grams and preserve lower order marginal constraint

**Power Low Rank Ensembles**
- Ensemble composed of unsmoothed $n$-grams plus other low rank matrices/tensors
- Alter lower order distributions by elementwise power

?
Generalizing KN to PLRE

**Kneser Ney**

- Ensemble composed of unsmoothed $n$-grams
- Alter lower order distributions by using count of unique histories
- Use absolute discounting to interpolate different $n$-grams and preserve lower order marginal constraint

**Power Low Rank Ensembles**

- Ensemble composed of unsmoothed $n$-grams plus other low rank matrices/tensors
- Alter lower order distributions by elementwise power
Key Requirements

• Marginal constraint must hold:

\[ \hat{P}(w_i) = \sum_{w_{i-1}} \hat{P}_{sm}(w_i | w_{i-1}) \hat{P}(w_{i-1}) \]

• Evaluation of conditional probabilities must be fast
Our Approach: Two Step Procedure

- **Step 1:** Compute discounts on powered counts such that marginal constraint holds. Each count gets a *different* discount.
Our Approach: Two Step Procedure

• **Step 1:** Compute discounts on powered counts such that marginal constraint holds. Each count gets a *different* discount.
Our Approach: Two Step Procedure

• **Step 1:** Compute discounts on powered counts such that marginal constraint holds. Each count gets a *different* discount.
Our Approach: Two Step Procedure

- **Step 2:** Take low rank approximation of discounted quantities such that marginal constraint still holds
Our Approach: Two Step Procedure

- **Step 2:** Take low rank approximation of discounted quantities such that marginal constraint still holds.
Our Approach: Two Step Procedure

- **Step 2:** Take low rank approximation of discounted quantities such that marginal constraint still holds.
Our Approach: Two Step Procedure

- **Step 2:** Take low rank approximation of discounted quantities such that marginal constraint still holds.

![Diagram showing the process of taking low rank approximations of discounted quantities.](image)
Why It Works

- Low rank approximations with respect to $KL$ preserve row/column sums
Why It Works

- Low rank approximations with respect to $KL$ preserve row/column sums
Why It Works

- Low rank approximations with respect to KL preserve row/column sums
Why It Works

- Low rank approximations with respect to KL preserve row/column sums
Why It Works

• Low rank approximations with respect to $KL$ 
  preserve row/column sums

• Therefore, discounting / leftover weight are preserved under the low rank approximation
Normalizer can be Precomputed

- Low rank approximations with respect to $KL$ preserve row/column sums
Normalizer can be Precomputed

- Low rank approximations with respect to KL preserve row/column sums

- Compute normalizers on sparse counts
Normalizer can be Precomputed

- Low rank approximations with respect to KL preserve row/column sums

- Compute normalizers on sparse counts

- No partition functions!
Marginal Constraint Holds

\[ \hat{P}(w_i) = \sum_{w_{i-1}} \hat{P}_{plre}(w_i \mid w_{i-1}) \hat{P}(w_{i-1}) \]
Generalizing KN to PLRE

**Kneser Ney**

- Ensemble composed of unsmoothed $n$-grams
- Alter lower order distributions by using count of unique histories
- Use absolute discounting to interpolate different $n$-grams and preserve lower order marginal constraint

**Power Low Rank Ensembles**

- Ensemble composed of unsmoothed $n$-grams plus other low rank matrices/tensors
- Alter lower order distributions by elementwise power
- Generalized discounting scheme: First compute discounts on powered counts, then take low rank approximation
Training Procedure
Training Procedure

count from corpus
Training Procedure

- Count from corpus

- Count from corpus
Use alternating minimization (EM) to compute low rank approximation with respect to KL [Lee and Seung 2001]
Training Procedure

- Because of ensemble representation, required rank is only about 100, even for billion word datasets

Use alternating minimization (EM) to compute low rank approximation with respect to KL [Lee and Seung 2001]
Test Time

KN Test Complexity: $O(n)$

$\quad n = \text{order}, K = \text{rank}$

PLRE Test Complexity: $O(nK)$
Test Time

KN Test Complexity: $O(n)$

$\begin{align*}
n &= \text{order}, K = \text{rank} \\
O(n) &\text{ complexity}
\end{align*}$

PLRE Test Complexity: $O(nK)$
Test Time

KN Test Complexity: $O(n)$

$\text{n = order, } K = \text{rank}$

PLRE Test Complexity: $O(nK)$
Outline

• Introduction

• Background on $n$-gram smoothing

• Our Approach
  • Rank
  • Power
  • Constructing the Ensemble

• Experiments
Experiments

• Evaluate on English and Russian

• Baselines
  • modKN – Modified Kneser Ney (back-off)
  • modint-KN- Modified Interpolated Kneser Ney
  • Other comparisons: Class-based models, Neural Networks, Hierarchical Pitman Yor
Small Datasets - Perplexity

- English-Small [Bengio et al. 2003]
  - 20K vocabulary
  - 14 million tokens

- Russian-Small
  - 77K vocabulary
  - 3.5 million tokens
Small Datasets - Perplexity

- English-Small [Bengio et al. 2003]
  - 20K vocabulary
  - 14 million tokens

- Russian-Small
  - 77K vocabulary
  - 3.5 million tokens

<table>
<thead>
<tr>
<th>Language</th>
<th>class KN</th>
<th>mod-KN</th>
<th>modint-KN</th>
<th>PLRE</th>
</tr>
</thead>
<tbody>
<tr>
<td>English-Small</td>
<td>119.7</td>
<td>104.55</td>
<td>100.07</td>
<td>95.15</td>
</tr>
<tr>
<td>Russian-Small</td>
<td>284.09</td>
<td>283.7</td>
<td>260.19</td>
<td>238.96</td>
</tr>
</tbody>
</table>
Small English Comparisons
## Small English Comparisons

<table>
<thead>
<tr>
<th>Model</th>
<th>Context Size</th>
<th>Perplexity</th>
</tr>
</thead>
<tbody>
<tr>
<td>mod-KN(4)</td>
<td>3</td>
<td>128</td>
</tr>
<tr>
<td>modint-KN(4)</td>
<td>3</td>
<td>116.6</td>
</tr>
<tr>
<td>infinity-gram HPYP [Wood et al. 2009]</td>
<td>infinity</td>
<td>111.8</td>
</tr>
</tbody>
</table>
## Small English Comparisons

<table>
<thead>
<tr>
<th>Model</th>
<th>Context Size</th>
<th>Perplexity</th>
</tr>
</thead>
<tbody>
<tr>
<td>mod-KN(4)</td>
<td>3</td>
<td>128</td>
</tr>
<tr>
<td>modint-KN(4)</td>
<td>3</td>
<td>116.6</td>
</tr>
<tr>
<td>infinity-gram HPYP [Wood et al. 2009]</td>
<td>infinity</td>
<td>111.8</td>
</tr>
<tr>
<td>PLRE(4)</td>
<td>3</td>
<td>108.7</td>
</tr>
</tbody>
</table>
## Small English Comparisons

<table>
<thead>
<tr>
<th>Model</th>
<th>Context Size</th>
<th>Perplexity</th>
</tr>
</thead>
<tbody>
<tr>
<td>mod-KN(4)</td>
<td>3</td>
<td>128</td>
</tr>
<tr>
<td>modint-KN(4)</td>
<td>3</td>
<td>116.6</td>
</tr>
<tr>
<td>infinity-gram HPYP [Wood et al. 2009]</td>
<td>infinity</td>
<td>111.8</td>
</tr>
<tr>
<td>PLRE(4)</td>
<td>3</td>
<td>108.7</td>
</tr>
<tr>
<td>LBL [Mnih and Hinton 2007]</td>
<td>5</td>
<td>117</td>
</tr>
<tr>
<td>LBL [Mnih and Hinton 2007]</td>
<td>10</td>
<td>107.8</td>
</tr>
<tr>
<td>RNN-ME [Mikolov et al. 2012]</td>
<td>infinity</td>
<td>82.1</td>
</tr>
</tbody>
</table>
Large Datasets - Perplexity

• English-Large
  • 836,000 types
  • 837 million tokens

• Russian-Large
  • 1.3 million types
  • 521 million tokens

• On 8 cores, PLRE (with optimal parameter settings) completes training on English-Large in 3.2 hrs and Russian-Large in 7.7 hours
Large Datasets - Perplexity

- English-Large
  - 836,000 types
  - 837 million tokens

- Russian-Large
  - 1.3 million types
  - 521 million tokens

<table>
<thead>
<tr>
<th></th>
<th>modint-KN</th>
<th>PLRE</th>
</tr>
</thead>
<tbody>
<tr>
<td>English-Large</td>
<td>77.90 +/- 0.20</td>
<td>75.66 +/- 0.19</td>
</tr>
<tr>
<td>Russian-Large</td>
<td>289.6 +/- 6.82</td>
<td>264.59 +/- 5.84</td>
</tr>
</tbody>
</table>

- On 8 cores, PLRE (with optimal parameter settings) completes training on English-Large in 3.2 hours and Russian-Large in 7.7 hours
Machine Translation Task

- English to Russian translation task (Language model is used as a feature in the translation system)

- Unlike other recent works, we use PLRE **instead** of modint-KN (not both)

- To deal with the non-determinism, the model is only trained once, using modint-KN. The same feature weights are then used for both PLRE and modint-KN
Machine Translation Task

- English to Russian translation task (Language model is used as a feature in the translation system)

- Unlike other recent works, we use PLRE instead of modint-KN (not both)

- To deal with the non-determinism, the model is only trained once, using modint-KN. The same feature weights are then used for both PLRE and modint-KN

<table>
<thead>
<tr>
<th>Method</th>
<th>BLEU</th>
</tr>
</thead>
<tbody>
<tr>
<td>modint-KN</td>
<td>17.63 +/- 0.11</td>
</tr>
<tr>
<td>PLRE</td>
<td>17.79 +/- 0.07</td>
</tr>
<tr>
<td>Smallest Diff</td>
<td>PLRE+0.05</td>
</tr>
<tr>
<td>Largest Diff</td>
<td>PLRE+0.29</td>
</tr>
</tbody>
</table>
Conclusion

• We presented a novel technique for language modeling called power low rank ensembles

• Consistently outperforms state-of-the-art Kneser Ney baselines
  • Effective for small context sizes
  • No partition function required

• Part of broader theme of exploiting connection between linear algebra and probability to develop new solutions for NLP
Thanks!