Contents

Neural Machine Translation of Rare Words with Subword Units

Paper Link: https://arxiv.org/abs/1508.07909

Authors: Rico Sennrich, Barry Haddow, Alexandra Birch (University of Edinburgh)

Publication: ACL 2016

Code: https://github.com/rsennrich/subword-nmt

Overview

This paper introduces Byte Pair Encoding (BPE) for neural machine translation, addressing the fundamental open-vocabulary problem where NMT models with fixed vocabularies struggle with rare and unknown words. The approach encodes rare words as sequences of subword units, enabling effective open-vocabulary translation based on the intuition that word classes can be translated through smaller linguistic units.

Key Problem Addressed

Neural machine translation faces the **open-vocabulary problem**:

1. **Fixed Vocabulary**: NMT models operate with fixed vocabularies
2. **Rare Word Handling**: Poor translation of rare and unknown words
3. **Out-of-Vocabulary (OOV)**: Unknown words replaced with generic tokens
4. **Translation Quality**: Significant impact on translation quality

Core Intuition

The paper is based on the insight that different word classes are translatable via smaller units:

1. **Names**

   - **Character Copying**: Direct character-by-character copying
   - **Transliteration**: Systematic character mapping across languages
   - **Examples**: “Smith” → “Smith”, “Tokyo” → “ ”

2. **Compounds**

   - **Compositional Translation**: Translate components separately
   - **Examples**: “basketball” → “basket” + “ball”
   - **German**: “Donaudampfschifffahrtsgesellschaftskapitän”

3. **Cognates and Loanwords**

   - **Phonological Transformations**: Sound-based mappings
   - **Morphological Transformations**: Structure-based mappings
   - **Examples**: “international” → “international” (similar across languages)

Technical Approach: Byte Pair Encoding

Original BPE Algorithm

BPE is a data compression technique that: 1. Starts with character vocabulary 2. Iteratively finds most frequent byte pair 3. Replaces pair with new symbol 4. Continues until desired vocabulary size

BPE Adaptation for NMT

1. Word Segmentation

```
# Start with word-level units
vocabulary = ["low", "lower", "newer", "wider"]

# Apply BPE merges
# Merge 1: "e" + "r" → "er" (most frequent pair)
# Merge 2: "w" + "er" → "wer"
# Result: ["low", "low" + "er", "new" + "er", "wid" + "er"]
```

2. Training Process

1. **Collect Statistics**: Count all symbol pairs in training data
2. **Find Most Frequent**: Identify most frequent adjacent pair
3. **Merge Symbols**: Replace pair with new symbol
4. **Repeat**: Continue until target vocabulary size reached

3. Segmentation Application

```
# Original: "lower"
# BPE segmentation: "low" + "er"
# Original: "unknown_word"
# BPE segmentation: "un" + "know" + "n_" + "word"
```

Implementation Details

Algorithm Steps

1. **Initialize**: Start with character-level vocabulary
2. **Count Pairs**: Count frequency of all adjacent symbol pairs
3. **Merge**: Replace most frequent pair with new symbol
4. **Update**: Update vocabulary and counts
5. **Iterate**: Repeat until desired vocabulary size

Vocabulary Size Selection

- **Trade-off**: Between compression and interpretability
- **Typical Sizes**: 10K-50K subword units
- **Language Dependent**: Varies by language characteristics

Special Handling

- **Word Boundaries**: Preserve word boundary information
- **Rare Characters**: Handle characters not in training data
- **Consistent Segmentation**: Deterministic segmentation

Experimental Results

Translation Tasks

- **WMT 2015 English-German**: +1.1 BLEU improvement
- **WMT 2015 English-Russian**: +1.3 BLEU improvement
- **Baseline**: Back-off dictionary approach

Comparison Methods

- **Character-level**: Pure character-based models
- **Word-level + UNK**: Traditional word-level with unknown tokens
- **Back-off Dictionary**: Post-processing with dictionary lookup

Analysis

- **Rare Word Translation**: Significant improvement
- **Overall Quality**: Better overall translation quality
- **Computational Efficiency**: Manageable vocabulary size

Advantages of BPE

1. Open Vocabulary

- **No Unknown Words**: All words can be segmented
- **Rare Word Handling**: Effective handling of rare words
- **Consistent Processing**: Unified approach for all words

2. **Language Independence**

- **Universal Approach**: Works across different languages
- **No Language-Specific Rules**: Data-driven approach
- **Scalable**: Applicable to any language pair

3. **Compression Benefits**

- **Reduced Vocabulary**: Smaller vocabulary than word-level
- **Meaningful Units**: Subwords often correspond to morphemes
- **Efficient Representation**: Good balance between granularity and efficiency

Practical Applications

1. **Neural Machine Translation**

- **Primary Application**: Originally designed for NMT
- **Quality Improvement**: Significant BLEU score improvements
- **Production Systems**: Widely adopted in production

2. **Language Modeling**

- **Reduced Vocabulary**: Smaller vocabulary for language models
- **Better Generalization**: Improved handling of rare words
- **Consistent Performance**: Stable across different domains

3. **Text Generation**

- **Open Vocabulary**: Generate any text without vocabulary constraints
- **Morphological Awareness**: Better handling of morphology
- **Multilingual Models**: Enables multilingual text generation

Limitations and Considerations

1. **Segmentation Quality**

- **Meaningless Splits**: May create meaningless subword units
- **Linguistic Validity**: Not always linguistically motivated
- **Over-segmentation**: May over-segment common words

2. **Hyperparameter Sensitivity**

- **Vocabulary Size**: Critical hyperparameter
- **Merge Operations**: Number of merges affects performance
- **Domain Dependence**: Optimal settings vary by domain

3. **Computational Overhead**

- **Training Cost**: BPE training requires computation
- **Segmentation Cost**: Runtime segmentation overhead
- **Memory Usage**: Subword vocabulary storage

Impact on NLP

1. Widespread Adoption

- **Standard Practice**: Became standard in NMT
- **Framework Integration**: Integrated into major NLP frameworks
- **Research Foundation**: Basis for many subsequent works

2. Subword Research

- **Inspired Variants**: Led to other subword methods
- **Theoretical Analysis**: Motivated theoretical studies
- **Improved Methods**: Foundation for improvements

3. Multilingual Models

- **Cross-lingual**: Enabled better cross-lingual models
- **Shared Vocabularies**: Facilitated shared subword vocabularies
- **Transfer Learning**: Improved cross-language transfer

Follow-up Work

1. Improved Algorithms

- **SentencePiece**: Language-independent implementation
- **Unigram Language Model**: Probabilistic approach
- **WordPiece**: Google's variant

2. Applications

- **BERT**: Used WordPiece (BPE variant)
- **GPT**: Uses BPE for tokenization
- **T5**: Uses SentencePiece with BPE

3. Analysis

- **Linguistic Analysis**: Studies of subword linguistics
- **Optimization**: Better hyperparameter selection
- **Evaluation**: Improved evaluation methods

Key Insights

1. Subword Effectiveness

Subword units provide an effective middle ground between character and word-level processing.

2. Language Universality

The approach works across different languages without language-specific modifications.

3. Compression Principle

Data compression principles can be effectively applied to linguistic segmentation.

4. Practical Impact

Simple algorithmic changes can have significant practical impact on system performance.

Why Read This Paper

BPE is essential reading because:

1. **Foundational Method**: Introduced fundamental approach to subword tokenization
2. **Practical Impact**: Directly improved translation quality
3. **Wide Adoption**: Became standard in modern NLP
4. **Simple Elegance**: Demonstrates power of simple, principled approaches
5. **Historical Importance**: Marked shift in how NLP handles vocabulary

Key Takeaways

1. **Open Vocabulary**: Subword units enable open-vocabulary processing
2. **Language Independence**: Universal approach works across languages
3. **Compression Inspiration**: Data compression principles apply to NLP
4. **Practical Benefits**: Significant improvements in translation quality
5. **Foundation**: Laid groundwork for modern tokenization approaches

BPE represents a pivotal moment in NLP history, demonstrating how adapting techniques from other domains (data compression) can solve fundamental problems in natural language processing. Its elegant simplicity and effectiveness have made it foundational to modern NLP systems, influencing virtually all subsequent work in neural language modeling and machine translation.