Frustratingly Easy Model Ensemble for Abstractive Summarization

Hayato Kobayashi
Yahoo Japan Corporation / RIKEN AIP

Introduction

Background

• Model ensemble is known to be effective for text generation
  • Drawing back: More computational costs (number of models’ times)
  • Many studies on compression of an ensemble model (Hinton+ 2015, …)
  • Drawing back: Cannot be easily applied to other models

Approach

• Selects a majority-like output from generated outputs in post-processing, inspired by the majority vote in classification tasks

Contributions

• Proposes a new model ensemble method (post-ensemble)
  • Simple: Implementation “frustratingly easy” w/o model code change
  • Fast: Computational time is enough for practical use (3.7 ms/sent)
  • Effective: Performance is better than runtime-ensemble
  • Proves a relationship to kernel density estimation based on vMF kernel

Preliminaries

Encoder-Decoder Model

• Conditional language model that learns an appropriate output sequence y given a sequence x from a lot of pairs (x, y)

Runtime-Ensemble

• Averages word prediction probabilities p in each decoding step
  Arithmetic mean (EnsSum)
  Geometric mean (EnsMul)

Post-Ensemble

Difficulty and Idea

• Difficulty: Majority output may not exist for text-generation
  • Text is a sequence of tokens, not a label
  • Idea: Instead of exact match frequency, use cosine similarity

Algorithm

• Select a majority-like output based on cosine similarity in post-processing
  • Generate an output on each decoder
  • No code modifications
  • Easily parallelizable

  Select the closest one to the other outputs by cosine similarity

Theoretical Analysis

• Post-ensemble is an approx. of KDE based on vMF kernel

Theory 1. The output of Alg. 1 with \( K(X, x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}x^T \Sigma^{-1} x} \) is equivalent to the maximization of the first order Taylor series approximation \( p_\theta \) of the kernel density estimator \( p \) based on the von Mises-Fisher kernel, i.e.,

\[
\hat{p}(X) = \max_{p(x)} \hat{p}(X|x)
\]

where the approximation error \( R^2 \) of the output \( p \) with respect to the true density estimator \( p, i.e., R^2 = \max_{p(x)} \hat{p}(X|x) - p(x), \) is bounded by

\[
R^2 \leq \frac{1}{2} \exp(\|\mu\|_2^2) \|\Sigma^{-1}\|_F^2
\]

(8)

where \( \mu = \max_{p(x)} \hat{p}(x) \) and \( \tau = \max_{p(x)} \hat{p}(x) \).

Kernel Density Estimation (KDE)

• Non-parametric method to estimate a probability density

\[
f(X) = \frac{1}{n} \sum_{i=1}^{n} K(X_i, x)
\]

von Mises-Fisher (vMF) Kernel

• Natural variant of Gaussian kernel to a unit hypersphere

• Compatible with cosine similarity

\[
K_X(x,y) = \frac{1}{2\pi \rho(\phi)} \exp(\cos(\phi(x,y))) \frac{1}{2}(\rho(y)) = a - k
\]

\[
= \left( \frac{1}{2\pi \rho(\phi)} \exp(\cos(\phi(x,y))) \frac{1}{2}(\rho(y)) \right)
\]

Effect of Number of Models

• May be saturated after 32 models
  • Computational time is reasonable
  • Ensemble time: 3.7 ms/sentence
  • Decoding time: 44 ms/sentence
  • Runtime-ensemble couldn’t be calculated with over 16 models

Effect of Model Preparation

• Random: Randomly initialize parameters
• Self: Use the last 10 epochs in a single run
• Hetero: Use 10 model structures
  • 8 models by replacing default settings with (2,3) layers, (250,500) LSTM dim., (250,500) embedding dim.
  • 2 models by replacing BiLSTM with an LSTM and BiLSTM with different merge
  • Bagging: Extract 80% from a training set

Discussion

Relationships to Existing Communities

• New category of algorithms for model ensemble
  • Main: Model selection in preprocessing, model average at run-time
  • New: Output selection in post-processing

• New category of tasks for hypothesis reranking
  • Main: Select one from the N-best hypotheses of a single model
  • New: Select one from the best outputs of multiple models

Future Work

• Learning of kernel functions (metric learning)
• Learning to rank outputs (list-wise reranking)
• Active learning to select a new model structure
• Boosting-like-ensemble extending bagging-ensemble