

Generative Deep Learning

Opportunity for Metova:

Build core competence in AI via basic Python and some exploration in machine learning and specialized Python modules

Build on present capability in cloud microservices and UI in order to productize AI models and services

Use case: generative documents for training

Produce novel documents which are plausible as real internet traffic

- correct spelling
- correct sentence syntax
- believable text semantics

'The main contributions of this paper are as follows. White and black pixels in the image indicate edge and non-edge regions, respectively. In the following, we approach this generic idea under a specific interactive framework. We combine a modern denoising Neural Network with Radiance Caching to offer high performance CPU GI rendering while supporting a wide range of material types, without the requirement of offline pre-computation or training for each scene. '

Neural nets are effective means of generating novel content by producing probabilistic models able to generate

For text generation recurrent structures are used which combine inputs with previous outputs, something similar to a recursive filter in digital signal processing

The models produced are of two general types:

[1] a sequence of characters formed according to a probability model trained to follow a character by character distribution

[2] a sequence of words formed according a probability model trained word by word to follow a syntactic distribution

Yes, but the Queen home way the at the placsarnd of the sat a it,' the
QUED INK Affece herge how eacushing ov

Yes, but did they rossan that may the sing,' sayeing, round queer thatled
thing's she went on, I'm she can wa

Yes, but first's word.' She was byself, if our it was en!' seemes you se
leasions of the doous rige out oldes

Yes, but the other--
Whereet heam who her not
one and it didn't knowlinge s

Alternative methodology:

- Read in 'documents' filtering citations, abbreviations, footnotes, etc.
- Remove 'stopwords' - the, and, a, this, etc.
- Apply 'stemming' - remove prefixes, suffixes associated with tense for exp.
- For each sentence represent words by their frequency of use in the sentence, divided by the frequency of the words over all documents –so emphasize distinctive words in the sentence over common words used in many sentences
- Form a document-term matrix A – each row is a sentence and the columns are all words used in all sentences – this is a sparse $N \times M$ matrix typically hundreds of sentences by thousands of terms
- Singular Value Decomposition $A = U * S * V^t$
- The main idea is that the rows of U correspond to the sentences, but each row is weighted sum of 'features' as opposed to words as in A and the rows of U are near or far from each other according to their semantic similarity - i.e sentences are near or far from each other based on meaning
- New documents are created by replacing some set of the original sentences by sentences which are very near in meaning
- Thus the generated documents have correct spelling and grammar and generally a similar