

高速フーリエ変換で畳み込み

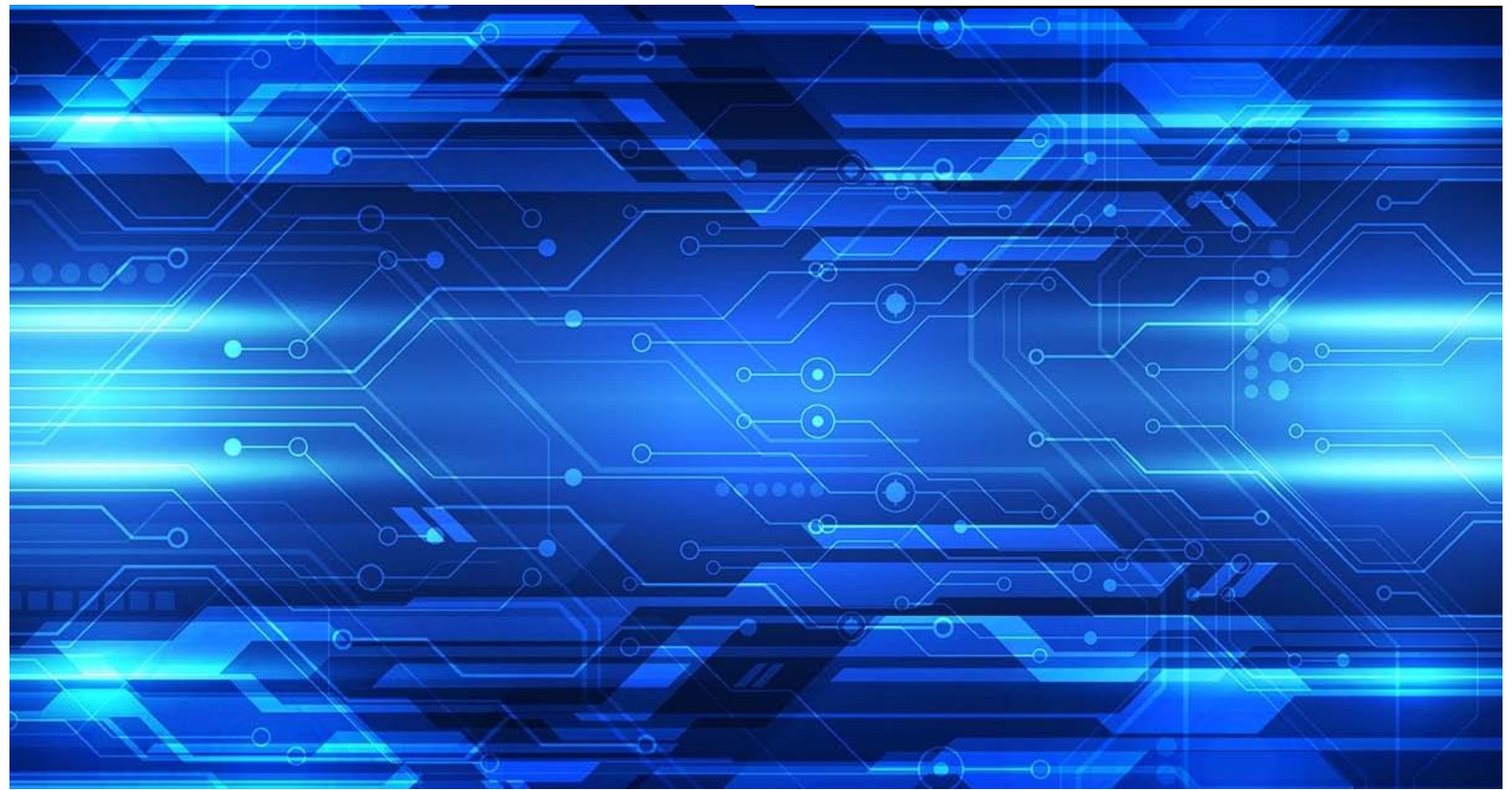
フーリエ変換と畳み込み

tmstk

ツイート

0

NHN TECHORUS
Tech Blog
AWS
Data Science
Tech
Event
Column



Topics

- 巡回畳み込みとは
- 離散フーリエ変換と巡回畳み込み
- 高速フーリエ変換で巡回畳み込みを高速化できる
- 高速フーリエ変換と巡回畳み込みの応用例
- まとめ
- 参考文献

こんにちは。データサイエンスチームのtmstkです。
前回までの記事（[1.離散フーリエ変換入門](#)、[2.高速フーリエ変換\(FFT\)入門](#)）では、離散フーリエ変換と、それを高速に計算するアルゴリズムである高速フーリエ変換を紹介しました。今回の記事では、離散フーリエ変換を用いて畳み込みを高速に計算する話を紹介します。

巡回畳み込みとは

離散フーリエ変換と相性のよい処理として、巡回畳み込みという処理があります。巡回畳み込みの定義は以下のとおりです。
 N を自然数として、 $x, y \in \mathbb{C}^N$ を N 次元ベクトルとする。ただし、 $x(t + N) = x(t), y(t + N) = y(t) (t \in \mathbb{Z})$ で x, y

を拡張し、周期 N をもつ整数値複素関数 $\mathbb{Z} \rightarrow \mathbb{C}$ と同一視する。このとき、 x, y の巡回畳み込み $x * y \in \mathbb{C}^N$ を

$$(x * y)(t) = \sum_{s=1}^N x(t - s)y(s) \quad (t = 1, \dots, N)$$

で定義します。
巡回畳み込みは、離散フーリエ変換と次のような関係があります。

離散フーリエ変換と巡回畳み込み

離散フーリエ変換と巡回畳み込みに関して、次が成り立ちます。

$$\mathcal{F}(x * y) = (\mathcal{F}(x)) (\mathcal{F}(y))$$

$(\mathcal{F}(x)) (\mathcal{F}(y))$ (ただし、 $(\mathcal{F}(x)) (\mathcal{F}(y))$ はベクトルの要素ごとの積（アダマール積）とします）。すなわち、巡回畳み込みの離散フーリエ変換は、離散フーリエ変換の要素ごとの積で計算できます。この性質は直接の計算により容易に証明することができます。

高速フーリエ変換で巡回畳み込みを高速化できる

高速フーリエ変換を使うと、巡回畳み込みの計算を高速化することが可能です。
巡回畳み込みは、定義にしたがって素朴に計算すると $(x * y)(t)$ を計算するのに N 回の掛け算と $N - 1$ 回の足し算が必要です。それを次元の数 N 回だけ繰り返し計算するので、全体の計算量は $O((N + (N - 1)) \times N) = O(N^2)$ になります。
一方で、先ほど離散フーリエ変換と巡回畳み込みの性質から

$$x * y = \mathcal{F}^{-1} \mathcal{F}(x * y) = \mathcal{F}^{-1} ((\mathcal{F}(x)) (\mathcal{F}(y)))$$

が成り立ちます。すなわち、巡回畳み込みを計算するには、 x, y の離散フーリエ変換を計算し、それらの要素ごとの積を計算し、その逆離散フーリエ変換を計算すればいいです。この方法だと、計算量が各ステップで $O(N \log N), O(N), O(N \log N)$ なので、全体の時間計算量は $O(N \log N)$ になります。
つまり、高速フーリエ変換を使うことで、巡回畳み込みの計算量が $O(N^2)$ から $O(N \log N)$ に削減できます。

高速フーリエ変換と巡回畳み込みの応用例

巡回畳み込みを高速フーリエ変換で高速化することの応用例として、機械学習における畳み込みへの応用を紹介します。
機械学習において、畳み込みとは以下のような処理です。いま、入力が1次元の場合を考えることにします。

畳み込みカーネルを $K \in \mathbb{R}^n$ とするとき、入力 $I \in \mathbb{R}^N$ に対するカーネル K による畳み込み $I \circledast K \in \mathbb{R}^N$ は、 $1 \leq t \leq N$ に対して

$$(I \circledast K)(t) = \sum_{s=1}^n I(t + s - 1)K(s)$$

で定義されます。（通常は $I * K$ と書かれますが、ここでは巡回畳み込みと区別するために $I \circledast K$ と書くことにします。）
畳み込みは、古典的な画像処理や最近流行りのディープラーニングでも用いられています。画像認識の本や

ディープラーニングの本に詳しいことが書かれています。
さて、この畳み込みは巡回畳み込みを使って計算することができます。例として、

$$I = (I(1), I(2), I(3), I(4))^T, K = (K(1), K(2))^T$$
 の場合を考えます。5次元ベクトル \tilde{I}, \tilde{K} を

$$\tilde{I} = (I(1), I(2), I(3), I(4), 0)^T, \tilde{K} = (0, 0, 0, K(2), K(1))^T$$
 で定めれば、巡回畳み込み $\tilde{I} * \tilde{K}$ は

$$\tilde{I} * \tilde{K} = (I(1)K(1)+I(2)K(2), I(2)K(1)+I(3)K(2), I(3)K(1)+I(4)K(2), I(4)K(1), I(1)K(2))^T$$

と計算されます。これの第1要素から第4要素までを抜き出したものが、畳み込み $I \circledast K$ と一致します。このようにして、機械学習における畳み込みを巡回畳み込みを使って計算することができます。

機械学習における畳み込みの計算量は $O(Nn)$ ですが、高速フーリエ変換と巡回畳み込みを応用することで計算量を $O(N \log N)$ にすることができます。これは、データのサイズ N が大きく、かつカーネルのサイズ n がデータのサイズ N に近い場合に有効です。
その他にも、多倍長整数の掛け算などを高速フーリエ変換を使って高速化できるようです。

まとめ

この記事では、巡回畳み込みを定義し、フーリエ変換の積が巡回畳み込みの離散フーリエ変換と等しいことを紹介しました。また、その応用として、機械学習における畳み込みを高速フーリエ変換を使って高速化できることを紹介しました。

参考文献

- Ian Goodfellow『深層学習』
- 山下幸彦他『工学のためのフーリエ解析』
- [アダマール積 - Wikipedia](#)
- [離散フーリエ変換を用いた多倍長乗算の話](#)

ツイート 0

#フーリエ変換#数学#深層学習

データ分析と機械学習とソフトウェア開発をしています。 アルゴリズムとデータ構造が好きです。

Recommends

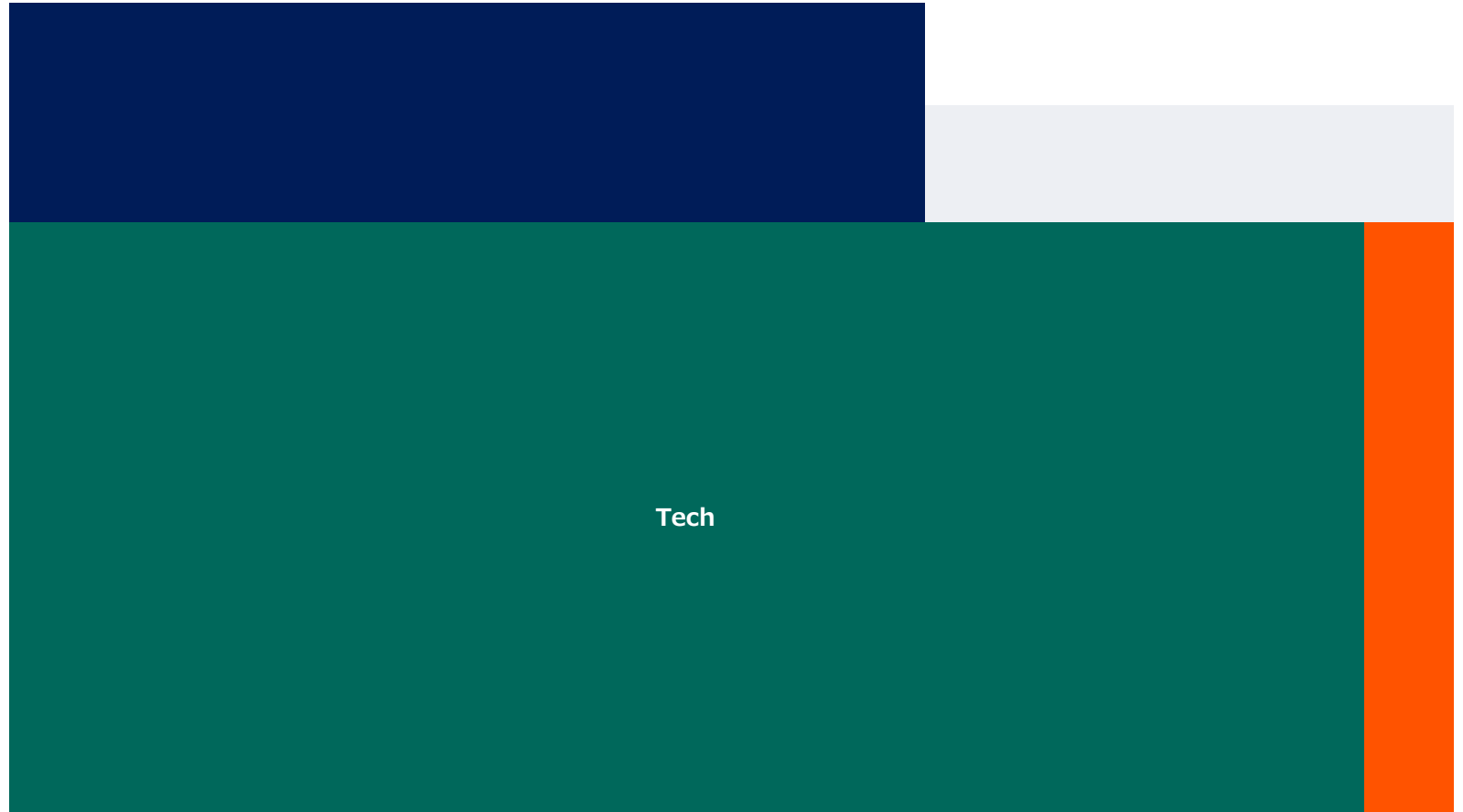
こちらもおすすめ

2018.7.26

高速フーリエ変換(FFT)入門

Data Science

S



GCPの利用料が安くなる|GCPの請求代行・運用代行・導入移行支援AWSの