

研究シーズ探索プログラム 研究課題別評価書

1. 研究課題名

遅延評価手法を用いた大規模統計システム構築法の確立

2. 研究代表者

篠崎 隆宏（東京工業大学 情報理工学研究所 計算工学専攻 助教）

3. 研究シーズ探索成果の概要

音声認識等において使用される大規模な統計システムについて、純粋関数型言語を用いることで改造・改良が容易な形でコンパクトに記述するための枠組みについて様々な検討を行った。具体的には、純粋関数型言語 Haskell を用いて重み付き有限状態トランスデューサ WFST を取り扱うためのデータ構造や探索関数を定義し、さらにその探索関数を用いて世界初の純粋関数型音声認識デコーダの実装を行い、3万語彙の大語彙連続音声認識タスクでの動作を確認した。従来の命令型プログラミング言語を用いて記述された認識デコーダが数万行のコードサイズであるのに対し、今回開発した純粋関数型デコーダはデータ構造の定義やコメント行等を含めてもわずか 400 行と非常にコンパクトに実装できることを実証した。計算コストに関しては今後の改良の余地があるものの、大規模な統計システムをコンパクトに実現するという点では目論見通りの結果が得られた。教育用の認識デコーダとしてはすでに十分有用であり、将来的に推進すべき研究シーズであることを確認できた。

4. 研究シーズ探索のねらい

本研究は統計的音声認識等において使用される大規模な統計モデルを用いたシステムの構築・改造・改良を容易にする、システムの新しい部品化および構築法の確立を目的とする。具体的には、重み付き有限状態トランスデューサ(WFST)の枠組みを遅延評価戦略に基づく純粋関数型プログラミング手法内に取り込んだ、時系列データ処理のための統計プログラミング手法を開発する。そして、大規模な統計システムをどこまで簡潔に記述でき、またどの程度までの規模のモデルを取り扱うことが可能であるかを明らかにする。

5. 研究シーズ探索の方法と成果

5.1 方法

大規模な統計モデルを用いた音声認識等の研究において、新しい統計モデルやその他様々な手法を試みるためには、しばしば学習や認識を行うソフトウェアの拡張が必要となる。しかしながら、これらのソフトウェアは複雑で大規模なものであり、その分野の専門家においても全体を理解し改造を行うことは容易とは言えず、研究遂行上のボトルネックとなっている。この問題に対するひとつのアプローチは、対象とする確率モデルの一般化を行いその一般化された確率モデルを扱うソフトウェアを開発することで、そのインスタンスである様々なモデルに対応することである。例えば近年応用が広がっている重み付き有限状態トランスデューサ(WFST) やベイジアンネット(BN) はこの例とみなすことが出来る。しかしながら、このアプローチではあらかじめ対象とされる事柄の範囲外のことを行おうとすると、やはり複雑なソフトウェアの改造が必要

になる問題がある。

他方、ソフトウェア工学の分野では純粋関数型プログラミングの研究が進み、純粋関数型言語およびその処理系が次世代のプログラミングパラダイムとして実用化されつつある。純粋関数型言語では関数に副作用が存在せず遅延評価との相性がよい特徴があり、高度に抽象化されたプログラムを記述することが可能である。また、関数の評価順序について処理系側の自由度が大きく、将来的には並列計算などにおいても有利と考えられている。そこで純粋関数型プログラミングを大規模統計処理ソフトウェアの開発に応用し、統計モデルの学習や認識処理のためのソフトウェアを柔軟かつコンパクトに記述する方法について、探索研究を行う。

5.2 成果

本研究では実証的に純粋関数型言語を用いて重み付き有限状態トランスデューサ WFST を取り扱うためのデータ構造や探索関数を定義し、さらにそれらデータ構造や探索関数を用いた大語彙連続音声認識デコーダを記述し、その可能性について検討を行った。純粋関数型プログラミングは従来の命令型プログラミングとは大きく異なった特徴があり、大規模な統計モデルの処理にどのように応用できるかは自明では無かった。このため、ソフトウェアの試作と、音声認識実験による評価とを交互に繰り返し、対象とする統計システムを小規模なものから次第に大規模なものへとスケールアップさせる戦略をとった。

本研究において開発を行った純粋関数型デコーダ「Husky」は、WFST 型の認識デコーダであり、プログラミング言語 Haskell を用いて実装されている。入力として AT&T 形式の WFST と HTK 形式の隠れマルコフモデル(HMM) の状態定義ファイル、音声特徴量ファイルのリスト、探索パラメタ等の設定ファイルを受け取り、音声認識を行うことが出来る。認識処理は1パス型のフレーム同期ビームサーチにより行われる。入出力はモナドを用いて実現しており、各音声特徴量ファイルを処理する毎に結果を出力する。プログラム上は音声特徴量ファイル全体を一度に読み込み探索部に渡すような構成となっているが、遅延読み出しを行うため、実際には探索の進行とともに開始フレームから逐次的に読み出される。表 1 に Husky のコード量を示す。純粋関数型プログラミングを用いた Husky は非常にコンパクトであり、データ構造の定義や入出力の為にコードを含めても 400 行に満たない。さらに探索や main 関数その他に限れば 120 行ほどである。これは従来の命令型言語を用いた音声認識デコーダのソフトウェアが数万行のオーダーであることと対照的である。

Table 1 Number of lines in Husky's source code

Category	Lines
Data structure definitions	43
IO functions	95
Main and search functions, etc.	119
Comments, blank lines	125
Total	382

純粋関数型言語では変数が存在しないため、構造データの操作について命令型言語とは異なった扱いが必要となる。例えば、純粋関数型言語では配列の一部の要素を変更することは不可能である。そのため元の配列の一部の値が更新された配列が必要な場合は、新たに配列全体を生成することとなり、配列の更新が非常に高コストとなる。他方線形リストのようなデータ構造への要素の追加は、表向きは全要素をコピーしつつも、実際には舞台裏でデータ構造の大部分を更新前のデータと共有することが出来、メモリや CPU の利用効率が高い。またサイズ平衡 2 分木等の木構造を用いる場合も要素の更新に対するコストが安価であり、純粋関数型プログラムにおいて多用される。配列もまったく用いられない訳では無く、一度初期化した後に参照のみを行う場合は定数コストで高速な参照を行えるメリットがある。これらを踏まえ、Husky

では WFST ネットワークや HMM 状態出力確率定義を配列、特徴量ベクトル系列を線形リスト、ビーム探索時の仮説候補を木構造により表現している。また探索時において、一般に HMM の同じ状態の同じ特徴量ベクトルに対する尤度が複数回使用されることから、計算効率上何らかのキャッシュ機構が必要となる。純粋関数型プログラミングでは処理系によっては自動的な最適化が期待できる場合もあるが、そうでない場合は配列要素の遅延評価を利用して計算結果をメモ化することや、あるいはそれと同等な機能を実現するライブラリ関数を利用することが考えられる。今回は後者の方法を用いた。

作成したソフトウェアは、最終的に 3 万語彙の大語彙連続音声認識実験により評価を行った。実験に使用した音声特徴量は MFCC12 次元と対数エネルギー、およびそれらのデルタ項とデルタデルタ項の計 39 次元である。音響モデルは 3000 状態の状態共有混合ガウス分布トライホンモデルであり、日本語話し言葉コーパス CSJ の学会講演音声より最小音素誤り基準 (MPE) により学習した。学習データ量は 254 時間であり、各状態の混合数は 32 である。言語モデルは CSJ の学会および模擬講演 6.8M 単語から学習したトライグラムモデルであり、辞書サイズは 30k である。WFST は、HMM 状態遷移、トライホン音素コンテキスト、発音辞書、トライグラム言語モデルを AT&T ツールキットを用いて合成して作成した。テストセットは男性話者による学会講演 10 講演からなる CSJ 評価セットである。Haskell 処理系には GHC ver 6.10.4 を用いた。不特定話者認識システムにおいて平均単語正解精度 81.1% が得られており、既存の T 3 デコーダと同等の良好な結果が得られた。他方、現時点での Husky の実装では、必要メモリ量が 40 Gbyte、実行時間が実時間の 62 倍程と大きい値となった。計算コストが高い理由の一つは、関数間でのデータの受渡しの背後でメモリの確保とガーベジコレクション (GC) が繰返されることである。ストリーム融合法の活用の他、グラフ構造の操作など汎用性のある関数の外部実装化やデータ構造の最適化などを行うことで、メモリと計算量は今後大幅に削減出来ると考えられる。

しかし、計算量の多さはあっても、Husky はそのソースコードが非常にコンパクトであることから教育用の認識デコーダとしてはすでに十分有用であり、将来的に推進すべき研究シーズであることを確認できた。尚、作成したソフトウェアは音声認識実験により評価を行っているが、WFST を用いた統計システムであれば他目的への応用は容易である。

6. 自己評価

私が専門とする音声認識は人工知能の分野の中でも独自の進化をとげた分野で、良くも悪くもガラパゴス的と言われたりします。特に大語彙連続音声認識を実現する既存のソフトウェアは大規模かつ複雑で、音声認識の専門家でさえも全体を理解して改良を加えることは必ずしも容易ではありません。このことは近隣分野と音声認識コミュニティを分断する障壁ともなっています。このような背景のもと、ソフトウェア工学の分野で研究・開発が進められている純粋関数型プログラミングを応用することで、如何に複雑な統計システムをコンパクトに記述するかという課題に取り組みました。純粋関数型プログラミングはこれまで私が慣れ親しんだ命令型のプログラミングとは相違に異なった特徴を持っており、また実際に開発を進め上で必要な情報がありそうでなかなか見つからなかったことなどが苦労した点です。最終的に、わずか 400 行に満たないコードで大語彙連続音声認識デコーダが記述可能ということを実証することが出来ました。計算効率については今後の改良が必要ですが、教育・プロトタイプ用認識システムとして、非常に有望であると考えています。今後は、本ソフトウェアを汎用の純粋関数型統計システム構築ツールキットとして発展させていきたいと考えています。

7. PO の見解

本研究は音声認識やバイオインフォマティクスで利用される統計モデルに基づいた大規模データ処理システムを効率よく構築する構成する手法の確立を目指したものである。WFST(Weighted Finite State Transducer)を利用した音声認識手法では、言語モデルや発音辞書等を WFST 形式に変換をし、合成し、探索ネットワークを生成する。このネットワークを用いて音声認識を行う。関数型言語 Haskell を用いて記述すると、これまでの手続き型言語 C を用いたものに比して2桁以上コンパクトに書け、認識精度も変わらないことを示した。

複雑な統計システムをコンパクトに記述可能パラダイムの提案が目的であったが、関数型言語の優位性を例題で示したことに留まっており、遅延評価にしても Haskell の評価機構を利用したに過ぎない。ソフトウェアの適用分野との連携は重要であるが、分野融合の観点では違和感がある。特定分野を情報技術で牽引するための、モデル、記述言語、処理系をその分野の研究者と情報科学者との連携で造り上げることの意義は大きい。

8. 研究成果リスト

(1)論文(原著論文)発表

今後投稿の予定です。

(2)特許出願

該当はありません。

(3)口頭発表

①学会

国内 1 件, 海外 0 件

篠崎 隆宏, 関嶋 政和, 萩原 茂樹, 古井 貞熙 “柔軟でコンパクトな純粋関数型デコーダの検討” 日本音響学会 2010 年秋季講演論文集, 1-Q-26, pp. 181-182, (2010.9)

②その他

国内 0 件, 海外 0 件

(4)その他の成果(受賞、著書、招待講演、特記事項等)

開発したソフトウェア(Husky)およびその解説を研究代表者(篠崎隆宏)ホームページで公開いたします。