

LF AI & DATA



モデル オープンネス フレームワーク (MOF) 仕様

LF AI & Data – Generative AI Commons

2024年12月17日

バージョン 1.0

この文書のステータス

これは**安定した文書**です。その内容はLF AI & Dataの内外で広範囲にわたるレビューを受けており、重大な問題は報告されていません。このドキュメントは[LF AI & Data Generative AI Commons](#)の成果物です。

1. はじめに

この仕様は、[モデル オープンネス フレームワーク \(MOF\) ホワイトペーパー](#)に基づいています。MOFは、オープンサイエンス、オープンソース、オープンデータ、オープンアクセスの原則に従い、機械学習モデルの完全性とオープン性を基に格付けを行うランク分類システムを定義しています。MOFでは、モデル開発ライフサイクルの特定のコンポーネントを適切なオープンライセンスで提供することを求めています。

この仕様は、準拠するために満たすべきモデル配布の要件のセットという形で、MOFの簡潔なバージョンを提供します。この仕様は、なぜそうするのかという理由よりも、何をすべきかという内容に焦点を当てています。MOFに含まれるさまざまな要素の背景や動機については、ホワイトペーパーを参照してください。

本文書におけるキーワード「MUST」、「MUST NOT」、「REQUIRED」、「SHALL」、「SHALL NOT」、「SHOULD」、「SHOULD NOT」、「RECOMMENDED」、「MAY」、「OPTIONAL」は、[RFC2119](#) で説明されているとおりに解釈されます。

2. MOF クラス

MOFでは、3つのクラス（クラスIII：オープンモデル、クラスII：オープンツリーリングモデル、クラスI：オープンサイエンスモデル）を定義しており、それぞれが前のクラスを基に構築され、モデルの完全性とオープン性の上昇レベルを表します。

各クラスには、特に記載のない限り、そのクラスに適合するには、すべてのコンポーネントが含まれ、かつ、モデルに適したオープンライセンスのもとでリリースされなければなりません。

オープンライセンスとは、あらゆる目的での無制限の使用、調査、修正、再配布を許可するライセンスです。

タイプに適したオープンライセンスとは、コード、データ、またはドキュメントといった対象のコンポーネントのタイプに適したオープンライセンスをさします。オープンソースライセンスとは、コードに適したオープンライセンスです。オープンデータライセンスとは、データに適したオープンライセンスです。オープンコンテンツライセンスとは、ドキュメントに適したオープンライセンスです。各コンポーネントタイプに適したオープンライセンスの推奨事項は、セクション4で示されています。

MOF クラス	含まれるコンポーネント	利用法
クラスIII：オープンモデル	<ol style="list-style-type: none"> 1. モデルアーキテクチャ 2. ファイナルモデルパラメータ 3. テクニカルレポートまたは研究論文 4. 評価結果 5. モデルカード 6. データカード 7. サンプルモデル出力（オプション） 	<ul style="list-style-type: none"> ・ 無制限の利用 （アクセス、使用、修正、再配布） ・ 製品またはサービスの作成 ・ 調整と微調整 ・ モデルの最適化
クラスII：オープンツリーリングモデル	<ol style="list-style-type: none"> 1. クラスIIIのすべてのコンポーネント 2. トレーニング、検証、およびテストコード 3. 推論コード 4. 評価コード 5. 評価データ 6. サポートライブラリおよびツール 	<ul style="list-style-type: none"> ・ トレーニングプロセスを理解する ・ ベンチマークの主張を検証する ・ 推論の最適化
クラスI：オープンサイエンスモデル	<ol style="list-style-type: none"> 1. クラスIIおよびIIIのすべてのコンポーネント 2. 研究論文 3. データセット 4. データ前処理コード 5. 中間モデルパラメータ 6. モデルメタデータ（オプション） 	<ul style="list-style-type: none"> ・ エンドツーエンドの分析と監査 ・ 類似モデルの再現 ・ データ調査と実験

3. MOF コンポーネント

以下では、上記のモデルのクラスに含まれる17のコンポーネントを定義します。これらは、MLモデルの開発プロセスのあらゆる側面における完成度とオープン性をカバーしており、トレーニングデータ、モデルアーキテクチャ、モデルパラメータ、評価ベンチマーク、およびドキュメントを含みます。

すべてのコンポーネントがすべてのクラスで必要になるわけではありません。以下の各コンポーネントのセクションではクラスを指定し、それが適用される各クラスに必要なコンポーネントを上記の表に記載しています。

すべてのコンポーネントを個別に配布する必要はなく、一部を組み合わせてもかまいません。例えば、評価結果は、独立した成果物として公開するのではなく、研究論文、テクニカルレポート、またはモデルカードに含めることができます。ただし、このような組み合わせは、そのコンポーネントまたは配布全体に対する LICENSE ファイルで対象とされているコンポーネントの種類に限定すべきです（第5項 LICENSE ファイルを参照）。

この仕様に対応するため、モデル配布には、セクション6で説明されているMOF構成ファイルという追加のコンポーネントを1つ含めるべきです。

モデルアーキテクチャ(III.1)-すべてのクラスに適用

モデルアーキテクチャには、MLアルゴリズム、ニューラルネットワークのレイアウト、接続パターン、活性化関数、その他のアーキテクチャ要素が含まれます。例としては、トランスフォーマー（GPT、BERTなど）、畳み込みニューラルネットワーク（CNN）、再帰型ニューラルネットワーク（RNN）、グラフニューラルネットワーク（GNN）などがあります。

モデルアーキテクチャは、研究論文、テクニカルレポート、またはモデルカードに完全に記載するべきであり（SHOULD）、OSI承認のオープンソースソフトウェアライセンスのもとでオープンソースコードとして配布する必要があります（MUST）。

ファイナルモデルパラメータ(チェックポイントとオプティマイザーの状態)(III.2)-すべてのクラスに適用

トレーニングされたモデルパラメータは、オープンライセンスの下で公開しなければなりません。ディープラーニングモデルの場合、トレーニングの主要な中間段階のチェックポイントと、最終的な最適化状態を記載する必要があります。少なくとも、最終的なモデルパラメータと最適化状態（該当する場合は）、圧縮または非圧縮のどちらであっても、TensorFlow、Keras、PyTorchなどの一般的なディープラーニングフレームワークと互換性のある形式、またはフレームワークに依存しないONNXファイル形式で配布する必要があります。

モデルパラメータ（重みとバイアス）は、Apache 2.0やMITのようなオープンソースソフトウェアライセンスには適さないデータです。モデルパラメータは、CDLA-Permissive-2.0のようなオープンデータライセンスの下で配布されるべきです。

モデルアーキテクチャとモデルパラメータは別々に配布されるべきです。なぜなら、この分離により、各コンポーネントを個別に調査、修正、再配布、使用することが可能になるからです。

テクニカルレポート (III. 3) - すべてのクラスに適用

テクニカルレポートは、ホワイトペーパー形式で作成することができます。このレポートは、モデルユーザーがモデルの性能、使用方法、および使用による影響を理解するために必要な情報を提供しますが、モデルを再現し、その結果を複製するのに十分な詳細情報を必ずしも提供するものではありません。

テクニカルレポートは配布物に含めるか、arXivのような恒久的なオープンアクセスプラットフォームで公開する必要があります。また、モデルカードからリンクを張るべきです。研究論文が提供されている場合は、テクニカルレポートを省略してもかまいません。

テクニカルレポートは、ドキュメントに適したオープンライセンス（理想的にはCC-BY-4.0）の下で配布する必要があります。

評価結果 (III. 4) - すべてのクラスに適用

モデルの評価に関する詳細な定量的指標と定性的結果を報告する必要があります。これらは、テクニカルレポート、研究論文、またはモデルカードに含めることができます。

テストでは、モデルの効率、正確性、パフォーマンス、公平性や偏りの評価、有害性、真実性などに限らず、あらゆる要素を評価することができます。

開発者は、業界標準ベンチマークであれ、開発したカスタムベンチマークテストであれ、ベンチマークテストの結果を含める必要があります。業界標準ベンチマークテストまたはテストスイートを使用した場合は、テストスイート名、テスト名、バージョン番号を結果とともに記載する必要があります。カスタムベンチマークを開発し

た場合は、コードであれ、テキストや画像を含むメディアであれ、検証のためにカスタムベンチマークのすべてを含める必要があります。

モデル評価の未加工の出力は、CC-BY-4.0のようなコンテンツに適切なオープンライセンスの下で配布する必要があります。

モデルカード (III. 5) - すべてのクラスに適用

モデルカードは、指標、使用方法のガイダンス、およびモデルの詳細を提供する。モデルカードは、モデルの詳細、意図する用途、要因、評価、リスク、およびモデルに関連する緩和策をカバーすべきです（[モデルカードによるモデルの報告](#)を参照）。

モデルカード自体は、ドキュメントに適したオープンライセンス（理想的にはCC-BY-4.0）の下で配布する必要があります。

データカード (III. 6) - すべてのクラスに適用

データカードは、データセットに関する要約統計やその他の詳細情報を提供しています。データカードは、機能、インスタンス、意図する用途、動機、収集プロセスに関するメトリクスを記述すべきです。

データカードはモデルカードと組み合わせてもかまいません。

データカードは、ドキュメントに適したオープンライセンス（理想的にはCC-BY-4.0）の下で配布する必要があります。

サンプルモデル出力 (III. 7) – すべてのクラスに適用

サンプル出力とは、テキストサンプル、画像、ビデオ、ソフトウェアコード、音声、3Dアセット、メタデータ、または予測や確率を含むモデルから生成されたその他の想定される出力です。

サンプル出力は推奨されますが、必須ではありません。省略してもかまいません。配布物にサンプル出力が含まれる場合、リリースとともに再配布できるように、著作権や制限を設けずに公開する必要があります。

特定の機密性の高いドメインについては、必要に応じて生成された例を匿名化または加工してもかまいません。モデル出力が著作権保護の対象とならない場合、その出力はライセンスなしで公開すべきであり、このことは LICENSE ファイルに記載する必要があります。

モデルユーザーが推論を実行した後に生成される実際のモデル出力は、MOFでは考慮されないことに注意してください。

トレーニング、検証、およびテストコード (II. 2) – クラスIIとIに適用

モデルのトレーニング、検証、テストのための完全なコード（モデルの構築、トレーニンググループ、ハイパーパラメータの選択、チェックポイントを含む）は、オープンソースソフトウェアとして配布されるべきです。あらゆるファインチューニングのコード、強化学習のコード、またはその他の方法でモデルのパラメータを変更するコード、あるいは最終的にモデルのパフォーマンスに影響を与えるアダプターを実装するコードを含める必要があります。

アプローチを説明するコメントは、コードに含める必要があります。理想的には、PythonコードのスタイルガイドであるPEP 8に準拠した形式で記載してください。また、トレーニング中に生成されたログファイルをリリースに含めることを推奨します。

トレーニング、検証、およびテスト用のコード自体は、OSI承認のオープンソースソフトウェアライセンスの下でリリースする必要があります。一方、ログはCC-BY-4.0のようなオープンコンテンツライセンスで利用できるようにすべきです。

推論コード (II. 3) – クラスIIとIに適用

推論を実行するためのコードには、推論中に必要となるデータの事前処理または後処理、および場合によってはモデルの最適化や外部ライブラリのような依存関係が含まれます。モデルのベンチマーク結果を完全に再現するために必要なコードはすべて含める必要があります。

推論コードは、OSI承認のオープンソースソフトウェアライセンスの下でリリースする必要があります。

評価コード (II. 4) – クラスIIとIに適用

モデル評価およびベンチマークに使用されるコードは、OSI承認のオープンソースソフトウェアライセンスのもとで配布する必要があります。

評価データ (II. 5) – クラスIIとIに適用

モデルがデータ（テキスト、画像、動画、音声、3Dデータなど、あらゆるメディア形式を含む）で評価される場合、その評価データは配布物に含める必要があります。

モデル開発者が広く普及している標準ベンチマークテストに依存している場合、それらのテストは配布物から省略してもかまいませんが、テクニカルレポート、研究論文、またはモデルカードに、テストのバージョンとともに記載する必要があります。

評価データは、CDLA-Permissive-2.0やCC-BY-4.0のような適切なオープンライセンスのもとで公開する必要があります。

サポートライブラリおよびツール (II. 6)

ー クラスIIとIに適用

モデルの開発過程で作成されたサポートコードライブラリ、ユーティリティ、ツールなどは、OSI承認のオープンソースソフトウェアライセンスのもとで配布する必要があります。これには、データローダー、ビジュアライゼーションコード、シミュレーション環境などが含まれます。既存のオープンソースツールやカスタムオープンソースツールの使用についても文書化する必要があります。

以下のツールおよびライブラリもすべて含める必要があります。:

- ・ モデル開発で使用されたソフトウェアライブラリおよびフレームワークと、バージョン詳細。
- ・ トークナイザー - テキストをトークン化するのに使用されるコード、およびトークナイザーのトレーニングに使用されるデータ。
- ・ ハイパーパラメータ検索コード - ハイパーパラメータのチューニングを自動化するコード。
- ・ 計算インフラストラクチャのコード - トレーニングの拡張に使用される特殊な計算インフラストラクチャのセットアップコード。
- ・ モニタリングのコード - モデル開発中に使用される実験、メトリクス、アーティファクトなどを追跡するためのコード。
- ・ コンテナ化ファイル - モデルを配布するための Dockerfile やその他のコンテナパッケージング。
- ・ フロントエンド/視覚化 - モデルの出力の上に構築されたウェブ/モバイルのフロントエンドや視覚化。

- ・ デプロイメントのオーケストレーション - モデルを本番環境にデプロイするための Infrastructure-as-Code テンプレート。
- ・ モデル統合コード - モデルを下流アプリケーションに統合するためのラッパーコード/SDK。
- ・ インタラクティブなデモ - Jupyter、Streamlit などを通じてホストされているモデルのインタラクティブなデモへのリンク。

おそらく、すでに使用されているほとんどのライブラリやツールには独自のライセンスがあると思われるが、モデル開発者が独自のライブラリやツールを作成した場合は、それらをOSI承認のオープンソースソフトウェアライセンスのもとで配布物に含める必要があります。

研究論文 (I. 2) - クラスIに適用

モデルの方法論、結果、分析を詳述した研究論文は、配布物に含めるか、arXivのような恒久的なオープンアクセスプラットフォームで公開する必要があります。また、モデルカードからリンクを張る必要があります。

研究論文は、ドキュメントにふさわしいオープンライセンス、理想的にはCC-BY-4.0で公開する必要があります。

以下の構成は推奨されるが必須ではありません: 概要、イントロダクション、関連研究、方法、結果、考察、結論、参考文献。

研究論文は、同じ情報を提供する詳細なテクニカルレポートに置き換えることができ、ドキュメントに適したオープンライセンス、理想的にはCC-BY-4.0で配布することができます。

データセット (I.3) - クラスIに適用

データセットには、事前学習、ファインチューニング、強化学習技術を用いたアラインメント、またはモデルの重みを変更するその他の方法に使用されるデータなど、あらゆる形式のモデル学習に使用されるデータであるトレーニングデータが含まれます。データセットには、モデルの検証およびテストに使用されるデータ、およびベンチマークテストで使用されるデータも含まれます。データセットコンポーネントには、トークン化されたデータセットが存在する場合も含まれます。

データは、テキスト、コード、画像、動画、音声、3Dオブジェクト、URI、トレーニング、検証、テスト目的で使用されるその他のデータなど、あらゆる形式またはメディアの組み合わせが可能です。

データセットには、あらゆるメタデータも含まれます。これには、ラベル、境界ボックス、キーポイントなどの注釈データから、権利帰属、ビットレート、解像度、モデル開発プロセスで使用されるデータセットに関連するその他のメタデータまで、あらゆるものが含まれます。

モデルの開発に使用されたデータセットは、著作権で保護されたデータとして、またはいかなる形態のライセンスのもとでも、パブリックドメインで提供されなければなりません。それらは、オープンライセンス、できればCC-BY-4.0またはCC-0.の下でリリースされるべきです。

プライバシーや機密性に関する共有の制限事項はドキュメント化されるべきです。前処理および後処理されたデータの両方が提供されるべきですが、データの前処理コードが添付されている場合は、生産者は代わりにオンラインで管理された生データセットへのリンクを提供してもかまいません。

データ前処理コード (I.4) - クラスIに適用

データ前処理コードとは、モデルのトレーニング、バリデーション、テストデータの事前処理、クリーニング、フォーマットに使用されるすべてのコードを指します。また、ファインチューニング データの変換に使用されるコードや、強化学習 (RLHF) のようなアラインメントタスクに使用されるコードも含まれます。その他、必要に応じてデータの取り込みに使用されるコード、フィーチャーエンジニアリング、データ拡張、トークン化などのデータ前処理コードも含まれます。

データの前処理コードは、OSIが承認したオープンソースソフトウェアライセンスを使用してリリースしなければなりません。

中間モデルパラメータ (チェックポイントおよびオプティマイザーの状態、ログファイル) (I.5) - クラスIに適用

クラスIモデルの場合、最終的なチェックポイントとオプティマイザーの状態に加えて、トレーニングの主要な中間段階におけるチェックポイントとオプティマイザーの状態 (該当する場合) とログファイルを含め、オープンライセンスのもとで配布しなければなりません。

中間モデルパラメータは、CDLA-Permissive-2.0などのオープンデータライセンスの下で配布されなければなりません。

モデルメタデータ (I.6) – クラスIIに適用

モデルに関する追加のコンテキストを提供できるメタデータの他の形式としては、モデルを作成する際に使用されたフレームワークのバージョンや、開発者によって提供されたカスタムタグや説明（モデルやデータの系統情報など）などがあります。このタイプのメタデータには特に要件やプロファイルはなく、開発者が出荷モデルに含めたいと思うものは何でも明らかになります。この情報は、特に複数のバージョンのモデルを使用したり、実験を行ったりする場合に、モデル管理に役立ちます。メタデータは、メタデータストアからエクスポートされたり、メタデータストアにロードされたりすることがよくあります。

モデルのメタデータは、モデルカード、研究論文、またはテクニカルレポートに含めることが可能です。

すべてのモデルメタデータは、CDLA-Permissive-2.0などのオープンデータライセンスで保護されていなければなりません。

モデル オープンネス構成ファイル – すべてのクラスに適用

MOF構成ファイルは、いかなる配布物にも含まれなければなりません。このファイルには、リリースに含まれるモデルコンポーネントと、各コンポーネントをカバーするライセンスが記載されています。

ファイル自体はオープンライセンスで配布されなければならない、CC-BY-4.0ライセンスで配布されなければなりません。

詳細は第6項を参照してください。

4. MOF 推奨ライセンス

各コンポーネントは、データ、コード、またはドキュメントの3つのうちのどれかです。下記の表は、各コンポーネントに使用すべき標準オープンライセンスを規定しています。ただし、同等のライセンスを使用することも可能です。

コンポーネント	タイプ	推奨オープンライセンス
データセット	データ	推奨：CDLA-Permissive-2.0、CC-BY-4.0 許容：ライセンスなしのものを含むすべて
データ前処理コード	コード	許容：OSI-approved
モデルアーキテクチャ	コード	許容：OSI-approved
モデルパラメータ	データ	推奨：CDLA-Permissive-2.0 許容：OSI承認済み、オープンデータライセンス
モデルメタデータ	データ	推奨：CDLA-Permissive-2.0 許容：CC-BY-4.0、オープンデータライセンス
トレーニングデータ	コード	許容：OSI承認済み
推論コード	コード	許容：OSI承認済み
評価コード	コード	許容：OSI承認済み
評価データ	データ	推奨：CDLA-Permissive-2.0 許容：CC-BY-4.0、オープンデータライセンス
評価結果	ドキュメント	推奨：CC-BY-4.0 許容：オープンコンテンツライセンス
サポートライブラリ及びツール	コード	許容：OSI承認済み
モデルカード	ドキュメント	推奨：CC-BY-4.0 許容：オープンコンテンツライセンス
データカード	ドキュメント	推奨：CC-BY-4.0 許容：オープンコンテンツライセンス

コンポーネント	タイプ	推奨オープンライセンス
テクニカルレポート	ドキュメント	推奨: CC-BY-4.0 許容: オープンコンテンツライセンス
研究論文	ドキュメント	推奨: CC-BY-4.0 許容: オープンコンテンツライセンス
サンプルモデル出力	データ または コード	ライセンスなし

5. ライセンスファイル

すべての配布には、モデルで使用されているライセンスを記述した LICENSE ファイルを含めなければなりません。LICENSE ファイルには、モデルに適用されるすべてのライセンスを含めることを推奨します。例えば、ソフトウェアが Apache 2.0 の下で、すべてのドキュメントおよびデータが CC-BY-4.0 の下でカバーされている場合、どのテキストがどのライセンスに属するかを区別するために、両方のライセンスのテキストを、ライセンスの見出しを含めて LICENSE ファイルにそのまま含めるべきです。

あるいは、配布物には、その配布物に含まれる異なるコンポーネントに結び付けられた異なるLICENSEファイルを含めることが可能です。各コンポーネントのLICENSEファイルは、対象となる

コンポーネントのベースディレクトリに置かれるべきです。異なるタイプのコンポーネントが結合される場合、LICENSEファイルには、結合される異なるタイプのコンポーネントに適用されるすべてのライセンスを含めるべきです。

MOF構成ファイルには、配布に含まれる各コンポーネントに対応する適切なLICENSEファイルへのパスが含まれており、コンポーネントごとのLICENSE方式と単一のLICENSEファイル方式の両方をサポートしています。

6. 構成ファイル

MOF構成ファイルは、JSON、YAML、AIBOM形式などの機械可読形式のファイルであり、以下の情報を含んでいます。

- **Framework:** このオブジェクトには、フレームワーク自体に関連する詳細情報が含まれています。これには、以下の必須属性が含まれます。
 - **name:** フレームワークの名前。
属性タイプは文字列です。
 - **version:** フレームワークのバージョン番号。
属性タイプは文字列です。
 - **date:** フレームワークのバージョンが公開された日付。
属性タイプは「YYYY-MM-DD」形式の文字列です。
- **Release:** このオブジェクトには、リリースされるモデルの詳細が含まれています。いくつかの属性があります。:

- **name:** リリース名。属性タイプは文字列です。
- **version:** リリース用のバージョン。これはパラメータ数、または以前のバージョンとパラメータ数が異なる同じモデルのバージョンとを区別するその他の識別子です。属性タイプは文字列です。
- **date:** リリース日。属性のタイプは「YYYY-MM-DD」形式の文字列です。
- **type:** モデルの特性、すなわち言語モデル、画像生成、音声生成、画像分類、統計的機械学習、またはその他の数多くのモデルの種類。属性タイプは文字列です。
- **architecture:** 採用されたモデルアーキテクチャ、すなわち、トランスフォーマー、Diffusion、GAN、NERF、VGG、Resnet、K-means、またはその他のモデルアーキテクチャ。属性タイプは文字列です。

- **treatment**: ファインチューニング、構造的アライメント、RLHF、またはその他のオリジナルモデルのパラメータを変更する治療など、トレーニング後のあらゆる治療。治療が適用されていない場合、この属性は空文字列となります。属性タイプは文字列です。
 - **origin**: オリジナルモデル、通常はこれが基礎モデルです。リリースでこれが基礎モデルでない場合、この属性には修正されたモデルの名前とバージョンが含まれます。属性のタイプは文字列、または基礎モデルまたは派生モデルでない場合は空欄です。
 - **producer**: 名前モデルの制作者または発行者は、企業、組織、グループ、個人である可能性があります。属性タイプは文字列です。
 - **contact**: モデル制作者または発行者のメールアドレス。型は文字列、形式はメールアドレスです。
 - **mof_class**: モデルオープンネスチェッカーによって生成されたリリースの適格なモデル オープンネス フレームワーククラス。属性タイプは整数です。
- **Components**: このオブジェクトには、モデル配布に含まれるコンポーネントの一覧と、各コンポーネントの詳細が含まれています。:
 - **description**: デフォルト値を使用したコンポーネントのテキスト説明は許容されます。標準コンポーネント以外の新しいコンポーネントを導入する際には、コンポーネントの説明を含めることが重要です。
 - **location**: ディストリビューション内のコンポーネントの場所は、ルートディレクトリへの先頭スラッシュ付きのUNIX形式のフルパスが必要です。属性タイプは文字列です。
 - **license**: コンポーネントで使用されているライセンスの SPDX 識別子。ライブラリやツールではよくあることですが、単一のコンポーネントに複数のライセンスが使用されている場合は、カンマ区切りのリストで提供する必要があります。ライセンスフィールドには、有効な SPDX ライセンス識別子を使用する必要があります。識別子は、こちらでご確認ください: <https://spdx.org/licenses/>。属性タイプは文字列です。
 - **license_path**: ディストリビューション内のコンポーネントのライセンスファイルの場所は、ルートディレクトリを示すスラッシュで始まるPOSIX形式のフルパスが必要です。複数のコンポーネントが同じライセンスファイルを指すことができます。コンポーネントが複数のライセンスを使用する場合は、ライセンスファイルにすべての使用ライセンスのテキストを含める必要があります。あるいは、カンマで区切って複数のライセンスファイルを指定することもできます。ただし、ライセンス属性で指定されたカンマ区切りのライセンス名のリストに対応している必要があります。属性の型は文字列です。

[モデルオープンネスツール \(MOT\)](#) を使用すると、モデル構成ファイルを簡単に生成することができます。

7. スコープ外

MOFは、AIの安全性（バイアス、公平性、信頼性を含む）、性能テスト、レッドチーム演習、セキュリティとプライバシー、モデル提供に関連するコンポーネント、モデルの由来など、以下の項目については対応する予定はありません。

参考文献

The Model Openness Framework (MOF) whitepaper:
<https://arxiv.org/abs/2403.13784>

The Model Openness Tool (MOT): <https://mot.isitopen.ai>

Model cards for model reporting:
<https://doi.org/10.1145/3287560.3287596>

OSI-approved licenses: <https://opensource.org/licenses>

Apache 2.0: <https://www.apache.org/licenses/LICENSE-2.0.html>

MIT: <https://opensource.org/license/MIT>

CC-BY-4.0: <https://creativecommons.org/licenses/by/4.0/>

CDLA-Permissive-2.0: <https://cdla.dev/permissive-2-0/>

謝辞

このドキュメントの大部分の内容は、MOFホワイトペーパーの著者であるMatt White、Ibrahim Haddad、Cailean Osborne、Xiao-Yang (Yanglet) Liu、Ahmed Abdelmonsef、Sachin Varghese、Arnaud Le Horsのほか、この仕様への貢献に対して特に称賛に値するのは、Arnaud Le Hors（編集者）と Cailean Osborne です。

ライセンス

この仕様は、クリエイティブ コモンズの表示4.0国際（CC-BY-4.0）ライセンスに基づき公開されています。参照：<https://creativecommons.org/licenses/by/4.0/>

A. MOF論文からの変更点

この仕様は、[2024年10月18日付のMOFペーパーv6](#)で定められた要件とほぼ一致するよう意図されていますが、一部の領域ではこれと異なる部分が存在しています。このセクションでは、それらの相違点を説明します。

- ・ 第3項：仕様では構成ファイルの使用を推奨していますが、必須ではありません（すなわち、MUSTではなくSHOULDです）。
- ・ 第3条 - モデルカード：この仕様では、データカードをモデルカードと組み合わせることが可能です。
- ・ 第5条 - LICENSEファイル：本仕様は、異なるタイプのファイルが組み合わせられたコンポーネントごとのファイルの場合をカバーしています。
- ・ 第6条：仕様では、構成ファイルは機械可読形式であることが求められていますが、JSON以外の形式の使用も許可されています。

B. Generative AI Commonsについて

Generative AI Commonsは、Linux FoundationのAI & Data Foundationにおけるコミュニティ主導のイニシアティブです。これは、生成AIにおけるオープンサイエンスとオープンソースの原則の推進に焦点を当てた、ベンダーニュートラルなフォーラムであり、オープン参加のイニシアティブです。Generative AI Commonsは、中立的なガバナンス、オープンで透明性の高いコラボレーション、教育を通じて、効率的で安全、信頼性が高く、倫理的な生成AIのオープンソースイノベーションの民主化、発展、普及を促進することに専念しています。

Generative AI Commonsの詳細、およびコミュニティへの参加に関する詳細とリンクについては<https://genaicommons.org> をご覧ください。

C. LF AI & Data Foundationについて

LF AI & Data Foundationは、Linux FoundationでグローバルなAI & データ テクノロジー インフラストラクチャの重要なコンポーネントをホストするグローバルな非営利財団です。

この団体には、業界が抱える課題の解決に貢献するプロジェクトや取り組みを特定し、参加者の利益となるよう貢献することを目的として、世界トップクラスの開発者、エンドユーザー、ベンダーを結集しています。

LF AI & Data Foundationの詳細については、<https://lfaidata.foundation/>をご覧ください。

 twitter.com/LFAIDataFdn

 linkedin.com/company/lfaidata

 www.youtube.com/@lfaidata

 github.com/lfaidata

LF AI & DATA

本記事について
この日本語文書は、The Model Openness Framework (MOF) Specificationの参考訳として、The Linux Foundation Japanが便宜上提供するものです。

英語版と翻訳版の間で齟齬または矛盾がある場合（翻訳版の提供の遅滞による場合を含むがこれに限らない）、英語版が優先されます。

翻訳協力：吉田行男