Agenda

• Opening Remarks
• Brig. Gen. Dave Kumashiro, USAF, Ret., - Director for Research and Analysis, National Security Commission on Artificial Intelligence (NSCAI)
• Dr. Jill Crisman – Principle Director, Artificial Intelligence, DoD R&E
• Mr. Tom Morton – Deputy Director for the Joint Common Foundation, Joint Artificial Intelligence Center (JAIC)
• Mr. Nicolas Chaillan – Chief Software Officer, SAF AQ
• Closing Remarks
Opening Remarks
Final Report Placemats
Artificial Intelligence (AI) Modernization

Jill Crisman, Ph.D. Principal Director for AI
Office of the Undersecretary of Defense: Research & Engineering

Approved for public release: distribution unlimited
An AI arms race has begun

“Whoever becomes the leader in AI will become the ruler of the world”
-- Vladimir Putin
September 4, 2017

In 2017, China announced its goal to become the world leader in AI by 2030

“Whoever leads in artificial intelligence in 2030 will rule the world until 2100”
--- Indermit Gill, Brookings Institute
January 7, 2020

Other nations, particularly China and Russia, are making significant investments in AI for military purposes, including in applications that raise questions regarding international norms and human rights.

These investments threaten to erode our technological and operational advantages and destabilize the free and open international order.

The United States, together with its allies and partners, must adopt AI to maintain its strategic position, prevail on future battlefields, and safeguard this order.

China and Russia plan to boost scientific cooperation with focus on artificial intelligence and other strategic areas

-- Hong Kong News
Jun 10, 2020

-- DoD AI Strategy, 2018
AI Capabilities

AI Modernization

• AI Assisted Analysis
  - Help analysts find information about the situation in a firehose of unstructured data
  - Help to identify influence operations on our military personnel by our adversaries

• AI Assisted Information Fusion
  - Help warfighters fuse information from multiple sources to manage the battlefield

• AI Assisted Reasoning
  - Help commanders with tactical, operational, and strategic planning and decision support

The AI assistive software applications can lead to AI Autonomy

Autonomy Modernization

• AI that senses, thinks, and acts autonomously with or without human supervision
DARPA’s Three Waves of AI

1980s: Hand-coded software
- **Expert knowledge or criteria and logical reasoning**
  - The first wave of AI is represented by expert knowledge or criteria developed in law or other authoritative sources and encoded into a computer program in the form of an expert system.
  - **Example:** Online tax preparation

2012: Deep Machine Learning
- **Machine/Statistical learning**
  - Second-wave AI technology is based on machine learning, or statistical learning, and includes voice recognition, natural-language processing, and computer-vision technologies, among others.
  - **Example:** Face-recognition technology

Future: Combination
- **Contextual adaptation**
  - Third-wave AI technology combines the strengths of first- and second-wave AI, and is also capable of contextual sophistication, abstraction, and explanation.
  - **Example:** Autonomous ships

DoD SW Teams are Adopting
- DoD Deep Learning Pilots
- Rapidly Adopt New AI
Deep Learning is 2nd Wave AI

- Deep Learning (a subset of machine learning ML) is AI’s latest advance
  - Deep learning (DL) trains a deep neural network (DNN), a software model of brain, to perform complex digital AI tasks
  - DNNs outperform the best hand-coded rule-based software for some tasks – first comparative public demonstration in 2012 on image classification
  - DNNs can continue to learn given high-quality feedback

- ML/DL has overtaken the AI science and technology (S&T) community
  - Within 3 years, DL was adopted by over 50% of the computer vision community as a means of creating AI software components
  - Recently, DL has been accelerating the natural language processing and reasoning AI tasks as well
  - DL is accelerating the creation of AI capabilities, e.g., object detection in images/video, entity and event extraction in text, speech transcription, and learning next-move policies for winning strategic games
  - This is accelerating the creation of new applications based on the corporate collected data

- ML/DL is democratizing software development
  - Once deep features are learned, that can be reused to solve many similar AI tasks
  - Data scientists now using ML platforms and frameworks to adapt DNN models
  - Federated data and ML can now allow data scientists at the forward edge to contribute to core model development – even when mostly disconnected

ML is a software development technology
“It is likely that the most transformative AI-enabled capabilities will arise from experiments at the “forward edge,” that is, discovered by the users themselves in contexts far removed from centralized offices and laboratories”

DoD AI Strategy, 2018

“The United States has consistently (and significantly) more cited AI conference papers than China over the last decade”

The 2021 Artificial Index Report, Stanford University
The development of artificial intelligence: Rapid transition thanks to Mobile, Cloud and Big Data

- **1950**: Mainframe computing
- **1980**: Client server
- **2000**: Intranet/Internet
- **2010**: Mobile-Cloud

**Programmable era**

**Intelligence era**

DoD is Here

Artificial Intelligence

source allianz via @mikequindazzi
AI Hubs: Democratized AI Innovation Platforms

• Discoverable AI problems
  - Democratized AI that are open for community development that is searchable – allowing the AI development community to find open problems and challenges

• Democratized training experiences
  - Community works together to build training data pool or game engine and experiences used to train DL or Reinforcement Learning (RL)

• Community test scenarios
  - Community works together to design the scenarios used for testing – includes needed tests for biases, effectiveness, efficiency, and reliability – continuously improving
  - AI engineering integrates this into the Dev*Ops process

• DoD shared test results
  - Full testing results among community solutions on various versions of training experiences and test scenarios – continuously updated
  - Allows others to see strengths and limitation of various approaches and select where they can make a difference

• Continuously improving
  - Training experiences, test scenarios, and AI are continuously evolving throughout the lifetime of the AI component in the user application
ML is a software development technique

• Need to add ML/DL to the SW Factories
  - Add ML experiences to source repositories
    - Unsupervised ML corpora
    - Supervised ML labeled datasets
    - Reinforcement ML on game engines
  - Add Model Zoo
    - ML/DL source code and trained models
  - Add ML/DL to DevSecOps

• Reach back to AI engineering
  - Hardening latest AI research for integration of ethical DoD applications
Service Software Teams

• Responsible
  - DoD personnel are responsible for the ethical development, deployment, and use of AI

• User Centered Design
  - Users are frequently involved in the design and assessment of the software they use in operations
  - As users continuously work with AI, they will build the appropriate level of trust and create better human-AI teaming

• User Feedback
  - Users can help adapt their workflows to new AI demonstrated capabilities
  - Users can discuss what explanations they need for AI
  - Users can provide annotations needed to train AI as part of their normal task

• Agile Development
  - Software is developed and continuously deployed in small increments
AI Scientific Research Areas

- **Expand the Domain of AI/ML**
  - e.g., cyber, EM, multi-modal situational awareness

- **High Performance AI**
  - Reduce dependency on datasets for DL, possibly through use of synthetic data
  - AI acceleration to allow more capable DL/DNNs to be trained

- **Robust AI**
  - Develop an understanding and testing for AI failure modes and decrease AI’s brittleness

- **Adversarial AI**
  - ML has vulnerabilities that can be exploited – research is needed to understand and reduce risks of AI

- **Next Generation AI**
  - Contextual awareness and switching

- **Reinforcement Learning and Game Play**
  - Decision support in real-world environments

- **Human-AI teaming**
  - Research to understand how to keep users engaged for appropriate teaming and the democratization of AI creation
AI Modernization

**Lead in AI Research**

_DoD AI Researchers 6.1-6.2_

The U.S. and our allies must continue to lead in AI scientific research – ensuring that the U.S. and our allies have the most effective, efficient, and ethical AI to win economically and militarily.

**Democratize DoD AI Innovation**

_DoD AI Engineering 6.3-6.4_

Engage the entire DoD AI development community, cleared industry, and allied partners in continuous creation, testing, and improvement of reliable, ethical, and affordable AI/DNN solutions.

**Accelerate AI Adoption**

_DoD Software Teams 6.5 – O&M_

Through user-centered development process and define and integrate AI components as trusted partners in mission applications – allowing for user adaptation at the forward edge and requirements feedback.

---

**Built on a Common Foundation**

Compute

Big Information

Data

Search

Workforce
Impact

• Trustworthy AI-enabled User Apps
  - As users are involved in developing AI-enabled applications, through the software factories, AI will become more effective and efficient and users will develop an appropriate level of trust in their AI partners

• DoD Culture of Community and Sharing
  - Building on one another’s source code/models builds much better AI software, much faster
  - Community development of training experiences, test scenarios, and open evaluations help to build assured AI

• Talent Development and Retention
  - AI talent will come to the DoD – drawn by the mission
  - But the DoD must allow them to use and keep developing skills to retain and grow their talent

• Common Foundation
  - Available DoD big data can be used for understanding ourselves and our adversaries and will allow deep learning to flexibly expand to new capabilities

• Lower costs in operations and maintenance
  - By democratizing AI development and building upon previous work, this approach will reduce redundancy, speed development, and lower costs
DoD AI Ethics Principles, 2020

• Responsible
  - DoD personnel remain responsible for the development, deployment, and use of AI capabilities

• Equitable
  - The Department will take deliberate steps to minimize unintended bias in AI capabilities

• Traceable
  - AI capabilities will be developed and deployed such that relevant personnel possess an appropriate understanding of the technology

• Reliable
  - AI capabilities will have explicit, well-defined uses, and the safety, security, and effectiveness of such capabilities will be subject to testing and assurance within those defined uses across their entire life-cycles

• Governable
  - DoD personnel will have the ability to catch and deactivate AI capabilities that demonstrate unintended behavior
Final Thoughts

Coming together is a **beginning**, staying together is **progress**, and working together is **success**.

- Henry Ford
DevSecOps – for AI/ML

Tom Morton – Deputy, Joint Common Foundation
Joint Artificial Intelligence Center

Catalyst, Partner, & Tool for the Transformation of DoD through AI

Why the JAIC?
- Accelerate AI adoption
- Support warfighting & business processes
- Bridge the gap
- Lead governance

JAIC Roles & Responsibilities
- Senior AI Official
- Maintain AI activity & impact accounting
- Industry & academic relationships
- Drive AI education and training
- Coordinate AI efforts across services, agencies, and components

What does the JAIC do?
- Enhances and enables AI at scale
- Identifies and drives AI Solutions
- Removes barriers to entry for DoD
- Provides foundational capabilities
- Shares lessons learned and best practices

https://www.ai.mil
Joint Common Foundation (JCF)

Enables DoD by building a common foundation of tools, frameworks, and standards.

Solutions

- Assess and integrate the AI/ML fabric and critical enabling capabilities/services across the enterprise to minimize duplication and leverage/modify existing capabilities to develop AI at speed of relevance
- Provides new/future enterprise capabilities that fill gaps for AI/ML infrastructure and platform requirements

Successes

- Developing and providing an initial operations capability for an IL4 data science environment with an Authority to Operate for data scientists, while prototyping the integration of multiple clouds – Azure and AWS
- Established the JCF GITLAB Source Code Management repository at IL4 for controlled unclassified information, where we now have the ability to migrate AI/ML source code into a centralized repository that DoD users can access
Machine Learning in ‘context’

Stages of the ML CI/CD pipeline


Representative example

Can be GPU intensive/ iterative/
MLOps

MLOps – the processes and practices combining data engineering, machine learning, and DevSecOps with the objective to develop, deploy, and maintain the capability throughout its lifecycle.

AI Journey — Key Issues

1. Business Requirements and Goal Definition
2. Data Collect, Process, Analysis and Features
3. Model Development, Evaluation and Preproduction Test
4. Model Deployment, Inference, Monitoring, and Integration
5. Business Application and Adoption

- Security and Ethical Concerns
- Potential Risks and Liabilities
- Complexity of Integration With Existing Infrastructure
Joint Common Foundation

Lower the Technical Barriers to AI and Enable a broad range of DoD Customers to integrate AI.

- Secure cloud enabled platform
- Share and reuse of data, code, tools, and models
- An environment to prototype, develop, test, and integrate AI/ML software.
- Enable users lacking infrastructure and/or technical expertise.
# JCF - Current Service Offering

**Primarily AWS & Azure AI/ML PaaS @ IL4**

## Shared Services/ PaaS

<table>
<thead>
<tr>
<th><strong>Azure Data Intake and Catalog</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>- Azure Data Factories</td>
</tr>
<tr>
<td>- Azure Blob</td>
</tr>
<tr>
<td>- Azure Databricks</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Cross Platform</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>- JCF Portal</td>
</tr>
<tr>
<td>- JCF Information Repository</td>
</tr>
<tr>
<td>- JCF Workspace Requests</td>
</tr>
<tr>
<td>- Customer Login</td>
</tr>
<tr>
<td>- JCF Workspace Central Common Development Tools</td>
</tr>
<tr>
<td>- SonarQube</td>
</tr>
<tr>
<td>- Custom Tools</td>
</tr>
<tr>
<td>- Data Card Catalog</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>AWS</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>- Data Intake and Catalog</td>
</tr>
<tr>
<td>- AWS Glue</td>
</tr>
<tr>
<td>- AWS S3</td>
</tr>
<tr>
<td>- AWS Lambda</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Common Development Tools</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>- JIRA</td>
</tr>
<tr>
<td>- Confluence</td>
</tr>
<tr>
<td>- GitLab</td>
</tr>
</tbody>
</table>

## Customer Enclave

<table>
<thead>
<tr>
<th><strong>Azure Containerized Capabilities</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>- Azure Databricks</td>
</tr>
<tr>
<td>- Azure Data Science Virtual Machine (DSVM)</td>
</tr>
<tr>
<td>- Azure Machine Learning</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Data Tools/ Pipelines</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>- Azure Data Box</td>
</tr>
<tr>
<td>- Azure Data Lake Storage</td>
</tr>
<tr>
<td>- Azure Blob Storage</td>
</tr>
<tr>
<td>- Azure Cognitive Services</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>AI/ML Tools Pipeline</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>- Azure Machine Learning</td>
</tr>
<tr>
<td>- Azure Databricks</td>
</tr>
<tr>
<td>- Azure Bot services</td>
</tr>
<tr>
<td>- Azure DSVM</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Data Storage</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>- Azure Blob</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Cross Platform</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>- Data Prep</td>
</tr>
<tr>
<td>- Model Training</td>
</tr>
<tr>
<td>- Model Deployment</td>
</tr>
<tr>
<td>- Model operations</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Containerized Capabilities</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>- AWS Elastic Map Reduce (EMR)</td>
</tr>
<tr>
<td>- AWS SageMaker</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Data Tools/ Pipelines</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>- AWS Glue</td>
</tr>
<tr>
<td>- AWS S3</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Data Storage</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>- AWS S3</td>
</tr>
</tbody>
</table>

### AI/ML CI/CD/DevSecOps Platform

<table>
<thead>
<tr>
<th><strong>AI/ML Tools Pipeline</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>- AWS Elastic Map Reduce (EMR)</td>
</tr>
<tr>
<td>- AWS SageMaker</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Data Storage</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>- AWS S3</td>
</tr>
</tbody>
</table>

**Capability and tooling as of 15-MAR 27**
A JAIC/JCF led development effort to build upon and where necessary, extend the current instantiation of the DoD Enterprise DevSecOps reference design to provide capabilities supporting AI/ML development, verification & validation, authorization, and operations.
JCF Cloud Neutral

Support for multiple “Personas”
- Data Engineers
- Data Scientists
- Capability Developers/ Integrators

Differentiated AI/ML Factories & Pipelines
- DATA – Dataset creation & understanding
- MODEL – Model training and evaluation
- APPLICATION – Hardening & Rapid Authorization

AI/ML Problem Domains
- Time Series
- Natural Language/Documents
- Digital Media
- Computer Vision ...

Recognizes & Leverages
- Not all CSP Services available on all fabrics
- Platform One (DevSecOps & Hardened Containers)
- Cloud One (Infrastructure)
- Other JAIC/ JCF customers & Initiatives

Attributes & Objectives
- Multi Cloud
  - Runs on AWS and Azure Gov Clouds
- Secure
  - Zero Trust
  - Service Mesh;
  - Continuous Monitoring;
  - Apps get CTF pre-approval
- AI/ML User Dev Env
  - JCF AI/ML Development Environment
- Automated
  - Maximizes automation through workflows/ pipelines
- End-to-End AI/ML DevSecOps
  - Datasets, Models, & Apps
- Edge Deployable
  - Can move AI/ML computing to the Data
- Flexible
  - Allows adding new/emerging tools over time
JCF Cloud Neutral - Architooon

- AI/ML Data Engineer
- AI/ML Data Scientist
- AI/ML Capability Developer

**Data Factory**
- nifi
- Kafka
- kibana
- jupyterhub
- TensorFlow
- Studio
- Anaconda

**Model Factory**
- Airflow

**App Factory**
- Flask
- Cypress
- Podman
- Shiny
- TensorFlow
- OpenGraf
- Anchore

**Bootstrap and Common Tools**
- (mostly from P1 Big Bang)

**Translated from Chinese to English**
- Data Factory
- Model Factory
- App Factory
- Bootstrap and Common Tools

**Kubernetes Container Orchestration**

**Cloud Deployment Abstraction**

**Grey Matter**
- Data Mesh
  - &
  - Service Mesh
  - Istio

**JCF Admin/SRE**

**JCF**
- AWS IL2, 4, 5, 6
- Azure IL2

**JCF Azure**
- IL2, 4, 5, 6
- C1D IL2

**JCF AWS**
- IL2, 4, 5, 6
- C1D IL2

**HPC & Edge**

**SC2S IL6 in planning**
## Container Hardening

### Data Factory Containers

<table>
<thead>
<tr>
<th>Product</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>NiFi – AI/ML (2)</td>
<td>Approved</td>
</tr>
<tr>
<td>Zookeeper (2)</td>
<td>Approved</td>
</tr>
<tr>
<td>Kafka</td>
<td>60-day Approval</td>
</tr>
<tr>
<td>Ckan</td>
<td>On Hold</td>
</tr>
<tr>
<td>PostgreSQL</td>
<td>Approved</td>
</tr>
<tr>
<td>redis</td>
<td>Approved</td>
</tr>
<tr>
<td>JupyterHub (4)</td>
<td>Approved</td>
</tr>
<tr>
<td>JupyterLab (5)</td>
<td>1 of 5 Approved</td>
</tr>
<tr>
<td>UBI (2)</td>
<td>Approved</td>
</tr>
<tr>
<td>solr</td>
<td>On Hold</td>
</tr>
</tbody>
</table>

### Model Factory Containers

<table>
<thead>
<tr>
<th>Product</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>AirFlow – AI/ML (6)</td>
<td>Prov Approval</td>
</tr>
<tr>
<td>redis</td>
<td>Approved</td>
</tr>
<tr>
<td>PostgreSQL</td>
<td>Approved</td>
</tr>
<tr>
<td>JupyterHub (5)</td>
<td>Approved</td>
</tr>
<tr>
<td>JupyterLab (5)</td>
<td>1 of 5 Approved</td>
</tr>
<tr>
<td>Rstudio (2)</td>
<td>In Hardening</td>
</tr>
<tr>
<td>Kafka</td>
<td>60-day Approval</td>
</tr>
</tbody>
</table>

### Portal Containers

<table>
<thead>
<tr>
<th>Product</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nginx</td>
<td>Approved</td>
</tr>
<tr>
<td>nodejs</td>
<td>Approved</td>
</tr>
<tr>
<td>Portal</td>
<td>Not Started</td>
</tr>
</tbody>
</table>

### App Factory Containers

<table>
<thead>
<tr>
<th>Product</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Python</td>
<td>Approved</td>
</tr>
<tr>
<td>Tensorflow-Serving</td>
<td>Approved</td>
</tr>
<tr>
<td>GitLab (11)</td>
<td>Approved</td>
</tr>
<tr>
<td>Cypress</td>
<td>Approved</td>
</tr>
<tr>
<td>Podman</td>
<td>Approved</td>
</tr>
<tr>
<td>Twistlock</td>
<td>Approved</td>
</tr>
<tr>
<td>Anchore (5)</td>
<td>Approved</td>
</tr>
<tr>
<td>OpenScap</td>
<td>Approved</td>
</tr>
<tr>
<td>Shiny</td>
<td>In Hardening</td>
</tr>
<tr>
<td>SonarQube</td>
<td>Approved</td>
</tr>
<tr>
<td>ClamAV</td>
<td>Approved</td>
</tr>
<tr>
<td>Flask</td>
<td>Not Started</td>
</tr>
</tbody>
</table>

---

*Note:*
- **J** = Hardened by JCF (21 total, 17 appr)
- **P** = Hardened by Platform One (30 total, all appr)
- **C** = Hardened in collab with AI2C (8 total, 1 appr)
Cloud Neutral Deployment

- Development Environment in C1D
- Able to deploy on demand
- Infrastructure and Configuration as Code

Possible Use Case
- “Clusters” deployable to C1D, JCF AWS, JCF Azure
- “try before you buy”

Possible Use Case
- Edge Deployment
JCF Cloud Neutral Portal

Common Tools
The following tools are available for all Factories:
- Grafana: Metrics viewer
- Kibana: Elastic Search viewer
- Gitlab: Repository viewer
- Mattermost: Team chat (coming soon)

Data Factory
The Data Factory provides the functions of data ingest and data labeling through a set of templates. A template is a user-defined and user-configured component that represents a building block to be used in a larger pipeline. The catalog of available data templates as well as access to the underlying tools is available below.

Tools
- NiFi: data orchestration and transformation
- JupyterHub: dataset IDE
- CKAN: data cataloging

Resources
- NiFi Templates stored in GitLab

Libraries
The following libraries are available in NiFi and JupyterHub:
- FeatureTools: feature engineering
- TSFresh: feature engineering (time series support)
- Snorkel: data labeling
- Kafka: data streaming

Model Factory
The Model Factory provides the functions of model generation and model training. Models are algorithms pre-defined by a subject matter expert. The results of the generation and training can then be viewed in the underlying tools. Previously loaded models and the underlying tools can be accessed below.

Tools
- AirFlow: model orchestration and training
- JupyterHub: modeling IDE
- TensorBoard: model analysis
- RStudio Server: R language IDE

Resources
- DAG Repository stored in GitLab

Libraries
The following libraries are available in Airflow and JupyterHub:
- TensorFlow: modeling platforms/algorithms
- ScikitLearn: modeling platforms/algorithms (time series support)
- Kafka: data streaming
- Anaconda: python/R data science platform including:
  - sklearn

App Factory
The Software Application Development Factory provides the functions of an application DevOps pipeline, including source code repositories, CI/CD pipeline orchestration, monitoring, single sign-on and AI/ML application development tools. The repository of existing applications can be accessed below.

Tools
- SonarQube: static code analysis
- Shiny Server: served R applications
- SW App: app repo, pipeline orchestrator

Libraries
The following libraries are available in the Gitlab CTF repository:
- Anchore: container vulnerability scanning
- Twistlock: container vulnerability scanning
- OpenSCAP: container vulnerability scanning
- SonarQube: code vulnerability scanning
JCF Cloud Neutral - Quick Tour
Questions
Backup
NiFi Data Factory Orchestrator
```python
def stage_4_model_training():
    # Airflow Stage 3 - Train
    def train_lda(data):
        # This function trains the LDA model
        # We setup parameters like number of topics, the chunksize to use in Hoffman method
        # We also do 2 passes of the data since this is a small dataset, so we want the distributions to stabilize.

        num_topics = 100
        chunksize = 300

        token = data['tokenized']
        token = token.apply(lambda x: x.strip('[]').replace('"', '\').split(','))
        dictionary = corpora.Dictionary(token)
        corpus = [dictionary.doc2bow(doc) for doc in token]

        dictionary = corpora.Dictionary(data['tokenized'])
        corpus = [dictionary.doc2bow(doc.strip('[]').replace('"', '\').split(',')) for doc in data['tokenized']]
        # corpus = [dictionary.doc2bow(str(doc)) for doc in data['tokenized']]
        # corpus = [dictionary.doc2bow(str(doc)) for doc in data['tokenized']]
        t1 = time.time()
        # Low alpha means each document is only represented by a small number of topics, and vice versa.
        # Low eta means each topic is only represented by a small number of words, and vice versa.
        lda = LdaModel(corpus=corpus, num_topics=num_topics, id2word=dictionary, alpha=2e-2, eta=5e-2, chunksize=chunksize, minimum_probability=0.0, passes=2)
        t2 = time.time()
        print('Time to train LDA model on ', len(df), 'articles: ', (t2-t1)/60, 'min')
        return dictionary, corpus, lda

    # download from azure
    data_from = "sqlserver-weapon-basart-trainable-feature"
    create_temp_dir(data_dir)
    download_from_azure(data_from, "trainable_feature.csv", data_dir)

    # data read
    path = data_dir + "trainable_feature.csv"
    df = data_reader(path)  # np konuş_card_path
    # fit
    dictionary, corpus, lda = train_lda(df)
    doc_topic_dist = np.array([])
    for data in lda[corpus]

    # save df, dict, feature mat, lda
    train_df_path = data_dir + "train_df.csv"
    dict_path = data_dir + "word_dict.sav"
    model_path = data_dir + "lda_model.sav"
    feature_path = data_dir + "feature_mat.txt"
    df, cove_train_df_path,
    np.savez(feature_path, doc_topic_dist)
    dictionary.save(dict_path)
    lda.save(model_path)

    # save data to cloud
```

Airflow Model Factory Orchestrator

DAG: basart_pipeline

Stage Diagram:
- stage_0
- stage_1_data_loading
- stage_2_data_preprocessing
- stage_3_feature_engineering
- stage_4_model_training
GitLab App Factory Orchestrator

Update get_lb_ip.py

16 jobs for master in 10 minutes and 48 seconds (queued for 11 seconds)

No related merge requests found.
Closing Remarks