AIVON Technical Documentations

Layer 2 plasma protocol for building decentralized video applications including the first decentralized Video Search Engine

Sze-Chin Lee
sq@aivon.io

U-Zyn Chua
uzyn@aivon.io

1st September 2018
https://aivon.io

Abstract

Artificial Intelligence Video Open Network (AIVON) is a decentralized blockchain protocol and ecosystem built on a consensus network of Artificial Intelligence (AI) computing resources and a community of human experts. AIVON generates normalized and enhanced metadata for video content, which forms the base data layer for the AIVON protocol that developers can use to build decentralized video applications including the first decentralized Video Search Engine. Developers can also build decentralized Applications (DApps) on top of the AIVON protocol. Platform participants can either run AI nodes or provide expertise and be rewarded with AIVON Tokens.

This paper proposes the technical implementation of AIVON which will be built using the Plasma protocol, as an Ethereum sidechain. Plasma is a layer 2 blockchain scaling solution designed by Joseph Poon and Vitalik Buterin which enables the creation of "child" blockchains anchored to the "main" Ethereum blockchain. As only periodic merkleized commitments (e.g. computations committed into merkle proofs) are broadcast to the Ethereum root blockchain, this design can allow for extremely scalable and low-cost transaction state updates, as well as flexibility in implementing our consensus algorithm to ensure fair assignment of worker nodes, accurate validation of metadata and proper distribution of earned token.

Table of Contents

1. Core Features

Following are the core features of AIVON:

- Allowing requesters to submit video to obtain normalized and enhanced metadata of that video.
- Enabling pooling of crowdsourced computing resources to create a decentralized artificial intelligence (AI) network for jobs processing.
- Allowing human to aid in the metadata verification and editorial process as well as continually train and improve AI datasets through Machine Learning (ML)
- Enabling all of the above in a decentralized manner, such that:
    - No central entity is able to influence the outcome of the data output.
    - All participants in the ecosystem should be able to perform their tasks independently.
    - Nodes that shares computing resource and workers must be rewarded. Rewards should be given in a decentralized and independent manner.
    - Jobs correctness must be able to validated up to a certain level of confidence, discouraging cheating and gaming of the system.
- The metadata generated include the following:
    - Time code of identified faces, objects,, images, text or scenes
    - Transcription (Optional)
    - Translation of Subtitles, closed caption or transcription (optional)
    - ContentGraph which is a vector of confidence score of various content safety attributes

2. Ecosystem Participants

There are 5 independent participants in the AIVON ecosystem:

1. Requesters
2. Computing resource providers
3. Human workers
4. Validators
5. Software developers

With the exception of Software Developers, all participants of the AIVON ecosystem will be required to stake AIVON tokens to a deposit pool to be eligible to participate.

2.1 Requesters

- Requesters submit videos to AIVON to obtain a normalized metadata of videos.
- Requesters pay for the job in the native AIVON token.
- Value Proposition
    - Metadata generated is normalized, trusted and open.
    - ContentGraph generated is similar to a MPAA rating but with normalized scoring for different content safety attribution.
    - Metadata can be consumed and reused by other softwares in a trustless manner.

2.2 Computing Resource Providers

- Providers run an open source software that shares the computer's computing resource onto AIVON
- The computing resource is being used for processing of videos through an AI model to generate metadata as well as the ContentGraph
- Value Proposition
    - Providers gain token in exchange for the provided computing resource and tasks performed.

2.3 Human Experts

- Human experts perform simple micro tasks to:
    - Ensure that the output from AI is correct and accurate
    - Rectify or edit the metadata
    - Provide additional metadata that cannot be easily generated by AI
- Relevant results can be fed into the AI network to teach and improve it.
- Value Proposition
    - Workers gain token in exchange for well performed tasks.

2.4 Validators

- Validators stake a significant amount of AIVON token and gain the right to perform the following tasks in accordance to consensus:
    - Assignment of job snippets to AI or HE nodes
    - Validation of the results submitted from AI/HE nodes.
    - Publication of final metadata and ContentGraph by validating the Result Claim submitted by the Requester.

- Value Proposition

    - Validators gain token for the job.
- Validator may lose a portion of the staked tokens if it is not performing the tasks in accordance to consensus.

2.5 Software Developers

- Developers play a role in the ecosystem by developing applications that interact with AIVON or utilize the AIVON generated metadata
- AIVON will be developing at least the following 3 decentralized applications based on the AIVON protocol:
    - Decentralized search engine based on the AIVON metadata layer
    - ContentGraph publishing service
    - Advertisement matching service
- Value Propositions
    - Developers can gain access and utilize the public video metadata layer and ContentGraph generated by AIVON to develop 3rd party DApps and commercial services.

3. Blockchain

3.1 AIVON Token

AIVON token is implemented as ERC20 token on Ethereum mainnet.

3.2 AIVON Network

AIVON runs on Ethereum chain to leverage its robustness, great ecosystem of Dapps, security and strong immutability features. Base layer of AIVON consists of smart contracts on Ethereum.

Further consensus rules unique to AIVON are done on Plasma Network, a layer-two scaling solution on Ethereum that allows for much higher transactional throughput and consensus algorithm that is independent of that of Ethereum.

4. Plasma

Implementation of AIVON on Plasma Chain is following plasma-mvp model from OmiseGO.[1] Plasma-mvp functions using a Unspent Transaction Output (UTXO) model that is similar to Bitcoin. Participants join and exit AIVON Plasma Chain by depositing into and exiting from a smart contract on Ethereum mainnet, also known as root chain.

4.1 Rootchain Contract

[1] https://github.com/omisego/plasma-mvp

Deposit

deposit() is a function on the smart contract allowing participants to bring in AIVON tokens into the child chain.

This function can be used by Requester to bring in AIVON tokens into AIVON Plasma to pay for processing fees, or for Validator to bring in the amount required for staking as Validator, similarly for AI and HE participants. Validator should make a deposit that is larger than or equals to the required staking amount.

Exit

startExit() To withdraw AIVON tokens from AIVON Plasma, participants would have to initiate an exit by calling the above function. This would begin the exiting process for a given UTXO and start a 24-hour period for other nodes to contest the exit in case a malicious user tries to exit with AVO that he does not own.

challengeExit() For anyone to challenge an exit by providing proof that an exiting attempt is not valid.

Depending on potential attack vectors and severity, **startExit()** may or may not involve collateral. If collateral is involved, successful exit challenge would involve both invalidating the exit attempts and also transferring of collaterals to successful challenger.

Block Submission

Validator periodically submits Merkle root of a AIVON block to the main contract. There is no need to submit every block created but only periodically every couple of hours.

Block Creation

Block creation is done within AIVON plasma chain by designated Validator. It includes a list of confirmed transactions in a block. See Data fields for more details.

5. Concepts and Data Formats

5.1 Block

An AIVON block consists of the following information:

- Block number
- Block hash
  - Hash of the content of the block
- Block validator
  - Address of the block validator
- Signature

    - Signature of block hash by block validator
- Next designated validator
    - Validator should be assigning next privileged Validator to create the next block
- List of transactions
    - List of all confirmed transactions in the block
    - This is inclusive also of newly minted AVO for Validator and AI nodes.
    - See Transaction section for more details.

5.2 Transaction Types

AIVON transactions may be one of the following types:

1. Job submission by Requester
2. AI node task assignment by Validator
3. HE node task assignment by Validator
4. AI task submission (encrypted) by AI node
5. HE task submission (encrypted) by HE node
6. Task Symmetric key revealed by Validator
7. AI job snippet metadata results published by Validator
8. HE job snippet metadata results published by Validator
9. Result claim submission by Requester
10. Result claim published by Validator
11. Transfer of AIVON Token using UTXO by any participant

5.3 Transaction Data Fields

S/N	Data Fields	Descriptions
1	AssetID	Hash value of transcoded Video Asset
2	SnippetID	Hash value of the content snippet
3	JobID	Hash value of the previous JobID and the SnippetID
4	JobType	3 types of job: Video Tag, Transcription and Subtitle Transcription
5	Encrypted content snippet minified url	Url to the encrypted content snippet hosted on Requester's local host

6	Content Symmetric Key	To encrypt content snippet and url
7	Task Symmetric Key	To encrypt metadata results generated
8	Recipient Address	Assigned Validator Node; Assigned AI/HE nodes
9	Sender Signature	
10	Confidence Range	Confidence range for each JobType which will trigger HE verification/correction. This is defined by Requester
11	Encrypted Metadata results	Metadata will be in JSON format
12	Inputs	A series of input AVO UTXOs that make up all the total outputs.
13	Outputs	An array of AIVON token outputs – recipients of AIVON tokens. Validator is also allowed to add an additional output without input as a block reward, thus making total output amount + block reward = total input amount.
14	Result Claim	Listing of SnippetID in correct order
15	Unused space	Unused space of a few bytes that allow for future protocol expansion

6. Transaction Workflow

6.1 Job Initialization Phase (Requester's Client)

1. Requester's client transcodes the video asset to an AIVON compatible video format. It then analyses the audio to determine appropriate time to segment the videos. It also generates a unique AssetID which is the hash value of the video asset.
2. Based on the segmented video, Requester extracts 1-5 image frames per second from the video and, together with the segmented audio to create the content snippet. For videos with fast moving frames such as sports, it is recommended for Requester's clients to extract 5 image frames per second.

$$c = \{c_1 , c_2 , c_3, \ldots, c_n\}$$

3. Requester generates a unique Content Symmetric Key for each content snippet and encrypt it.
4. Requester generates 2 other unique IDs:
   - SnippetID: Hash value of the content snippet

$$SnippetID_n = h(c_n)$$

$h = hash\ function$

- JobID: Hash value of the previous JobID and the SnippetID. This will enable verification of the sequence order of the validated results.

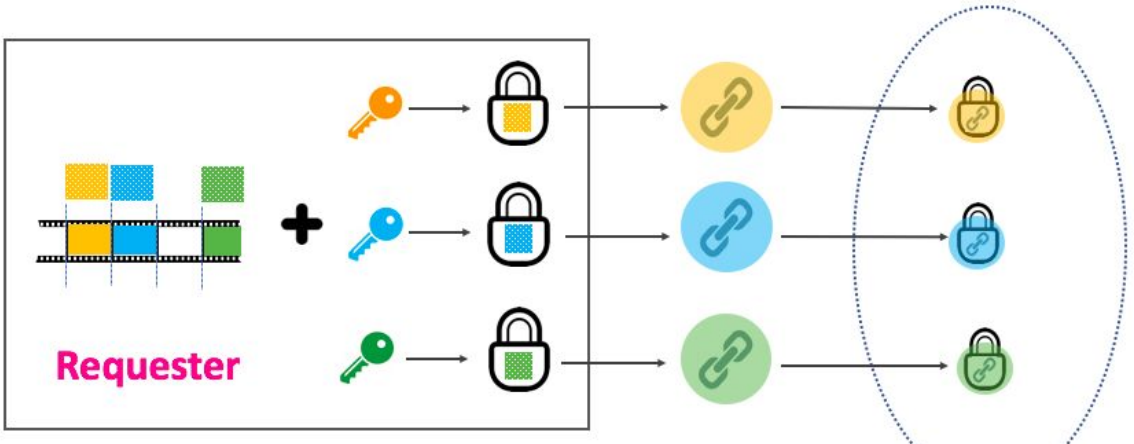
$$JobID_n = h(SnippetID_n, JobID_{n-1})$$

$h = composite\ hash\ function$

5. Generate a minified download url to the content snippets hosted on the Requester's client local host and encrypt the shortened url with same Content Symmetric Key.
6. Submit the signed content snippet url in a randomized sequence to AIVON together with AssetID and JobID

Notes:

- *(2) prevents any bad actors from stealing content by ensuring that no actual video clips are submitted to AIVON; it also reduces bandwidth requirement in terms of transmission. (4) ensures that only assigned AI/HE nodes have access to the content*
- *In the next version, AIVON would allow Requester to submit a job with their own specific trained AI model but still tap on AIVON compute and human resources. This will supersede the standard AIVON trained AI models.*

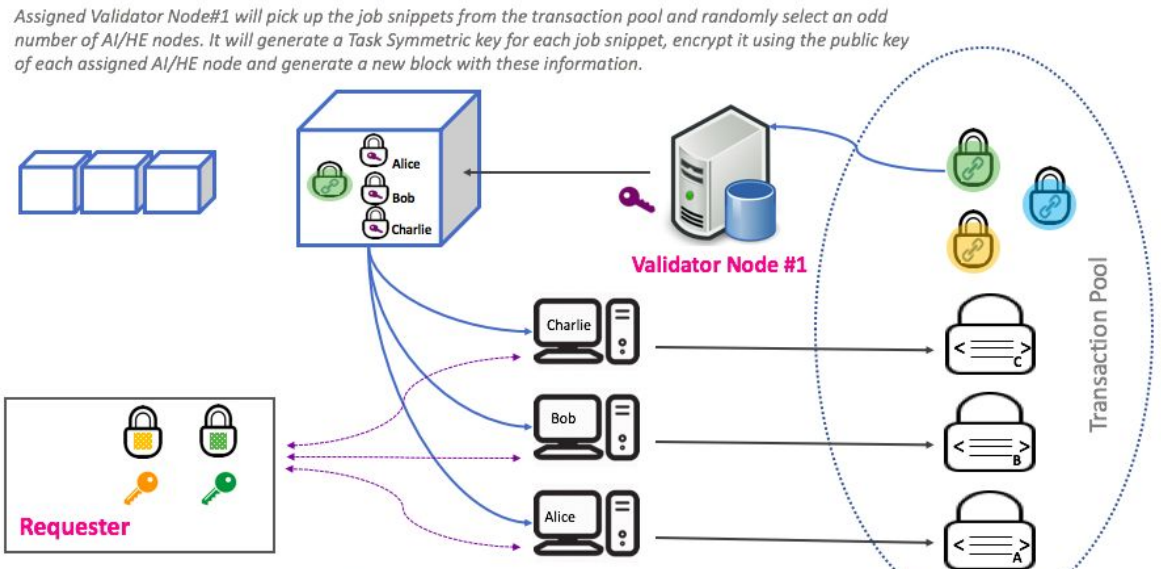


Requester will segment the videos into snippets and generate a Content Symmetric key for each work snippets. The extracted media files for each video snippets are encrypted and hosted locally. Requester will encrypt the minified url using the same Content Symmetric key and submit to the transaction pool for each content snippets.

6.2 Task Assignment & Execution Phase (Validator)

1. Requester submits job snippet url to the unconfirmed transaction pool as individual transactions

2. The assigned validator - Validator #1 will pick up the job snippets from the transaction pool, selects an odd number of AI nodes from pool of available AI Nodes and create a new block with the task assignments.
3. Validator #1 generates a Task Symmetric Key for each job snippets and publishes it encrypted with each assigned AI/HE node's public key along with each encrypted snippet url during task assignment.
4. Seeing the task assignment recorded in the blockchain, the Requester will use the public key of the AI/HE nodes to encrypt the symmetric key of the encrypted work snippets and its download url, and publish to the unconfirmed transaction pool.
5. AI/HE nodes can use its private key to recover the symmetric key that can then be used to decrypt the URL, download and decrypt the content snippets from the Requester.
6. AI/HE nodes perform the task and generate the metadata.
7. Metadata generated is encrypted with the Task Symmetric Key, signed by the AI/HE node and submitted to transaction pool.



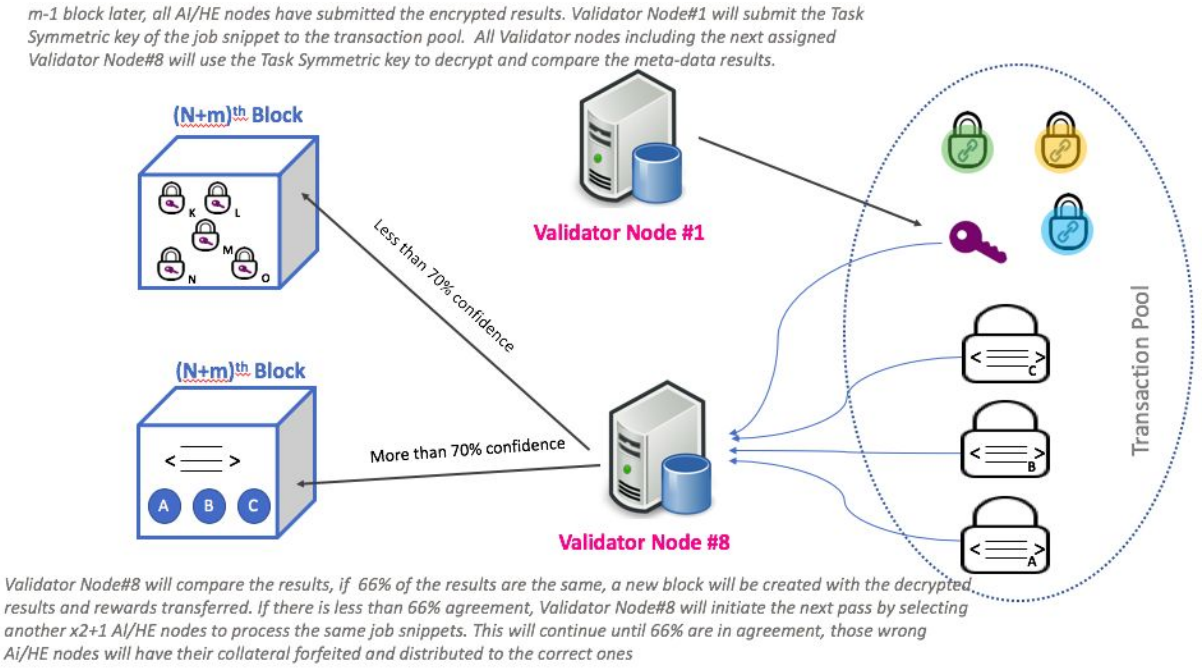
6.3 Result Validation by AI Nodes ("Proof of Intelligence")

1. A few blocks later (e.g. m-1 block later), all AI nodes have completed the tasks and submitted the encrypted results.
2. Validator#1, who has the Task Symmetric key to the job snippet, reveals the key by submitting a transaction to the transactions pool.
3. All Validators that are not assigned to create a block, see in the pool that all assigned tasks have completed (or have timed out) and that Task Symmetric Key is also available, now decrypts the metadata and perform validation.

4. Validation is performed by ensuring that majority of the result submitted for the same assigned tasks are the same. Majority consensus is achieved if at least 66% of the results are the same. If they are, the newly assigned Validator#8 publishes the result onto a confirmed block, awarding each of the AI nodes the reward for the work done, and also attaching "Proof of Intelligence" done by each AI nodes.
5. AI nodes that produce results that are different from the majority, hence not accepted, will be penalized by forfeiting a portion of its stakes.
6. If there is no majority, Validator Node#8 will initiate the next pass by assigning another group of the AI/HE nodes to process the same work snippets. The number of new AI/Nodes assigned will increased based on the formula

$$Number\ of\ new\ nodes = Previous\ number\ of\ Nodes * 2 + 1$$

7. The staked tokens in the first batch of AI/HE nodes will be locked up, pending the validation outcome of the next pass produced by next batch of assigned AI/HE nodes. This process will repeat until a majority outcome can be achieved.
8. Those wrong AI/HE nodes will have a portion of their staked collateral forfeited and distributed to the correct ones



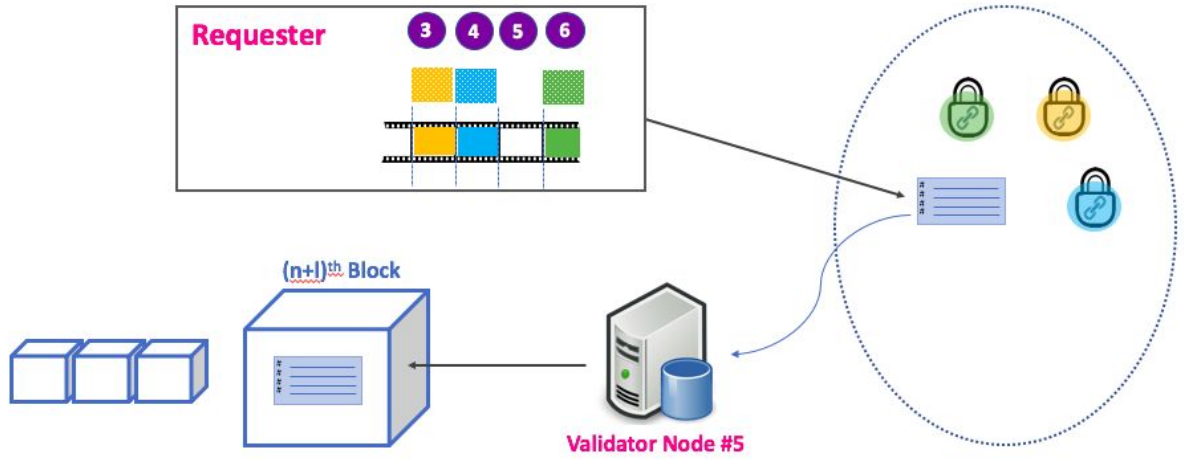
6.4 Result Verification & Correction by HE Nodes

1. If the confidence score of each meta-tag falls within a confidence range defined by the requester, it will trigger the result verification & correction by the HE nodes.

2. The assigned validator will randomly assign the verification task to an odd number of HE nodes from pool of available HE Nodes and create a new block with the task assignments.
3. The subsequent workflow is similarly to what was described in section 6.2 (3)-(7), however the tasks will be performed by humans.
4. For low skilled task which will yield a finite response, results are only fully accepted if at least 66% of all HE nodes provide the same results, i.e. 66% agreement. If the agreement is short of 66%, another pass would be initiated by Validator and number of HE nodes will increased by x2+1.
5. For high skilled task such as transcription and translation, it will follow the sequence below:
    a. Assigned HE nodes will perform the task which will likely yield different results
    b. Validator assign a group of HE review committee nodes (to rank and vote the results of the 1st pass according to 3 buckets: “unacceptable”, “acceptable”, “best”. Results are accepted if at least 66% of the HE review committee nodes vote for the same 1st pass result as the best.
    c. If the agreement is short of 66%, another pass would be initiated, repeating (a) and (b) and increasing the HE review nodes by x2+1.

6.5 Metadata Publication Phase

1. Requester sees that all the content snippet results are now published on the blockchain.
2. Requester submits the snippet sequence order claim to the transaction pool. The claim is essentially a sequence of the JobID and SnippetID in the correct order.
3. Assigned Validator #5 for this block processes, validates that the sequence is valid, and publishes the claim together with its unique hash to the blockchain, thereby completing the whole process for the particular video asset.
4. To encourage participants to make the processed metadata results public, when a Requester submits a job to AIVON, it will pay for a job fee as well as a refundable private fee.
5. Requester gets rewarded by successful video claim with the private fee refunded to encourage Requester to make its processed metadata public for the good of the community.
6. If a Requester wishes to keep its metadata private, it would only need to perform the aggregation privately on the client without submitting the claim with snippets sequence. The private fee will then be distributed to the participants involved in the job processing.



Requester sees that all its job snippet results are published on the blockchain, it will submit a snippet sequence order claim to transaction pool. The claim is essentially a sequence of the snippetID in the correct order. Assigned Validator Node#5 will validate the sequence order and if it's correct publish the sequence order to the blockchain.

7. Consensus

7.1 Validator Designation

After publishing a new block, Validator must assign the next Validator using a deterministic but random algorithm.

The chosen algorithm must satisfy the following properties:

1. It must be sufficiently random
2. It must be sufficiently non-exploitable
3. It must be deterministic
4. It must consider the amount of staked AIVON per node

7.2 Potential Exploits And Issues

1. Validator is able to deliberately publish blocks that only benefit themselves, e.g. that continuously nominate themselves or nodes that they control for subsequent block publishing.
2. Nominated validator goes offline or does not publish a block and holds up AIVON blockchain.

Random Number Generator

Proposed random number generator based on Vitalik's paper on randomness on blockchain[2], is to use a hybrid private randomness approach to minimize the exploitability of randomization algorithm.

[2] https://vitalik.ca/files/randomness.html

A good random number could be generated in the following manner:

$$R = h(\ e(n)\ ,\ a(n-1)\ ,\ a(n)\ )$$

h = composite hash function which serializes the multiple inputs before hashing[3]
e(n) = current block hash from the Ethereum Mainnet
a(n-1) = previous AVION block hash
a(n) = current AIVON block hash

Validator Timeout

To address the potential exploits and issues, nominated Validator must respond by a fixed amount of time. The chosen timeout must be sufficiently long that a moderately powerful Validator node has sufficient time to perform the necessary hash computation in order to publish a block, but must be sufficiently short that a Validator should not be able to wait until for multiple Ethereum mainnet blocks to pass to select one that is the most favorable for itself.

If a Validator fails to publish a block within a timeout period, next validator is determined by the Validator Nomination Algorithm below.

Proof of time pass must also be submitted by next+1 Validator in the block to prove that the last nominated Validator fails to act within a stipulated timeframe. This proof of time is submitted as block hash and block timestamp of parent chain (Ethereum Mainnet).

Validator Nomination Algorithm

The next Validator can be assigned using the following algorithm[4]:

$$Validator\ (n+1) = R\ mod\ N$$

N = total number of active Validators during the current period
mod = modulo operation
R = a random number generated using the method discussed above

The algorithm will take the Validator at that index as the next Validator but keep randomly picking Validators until it finds one that has not already been used in the current period

Validators' Stake

All Validator nodes are required to stake a significant amount of AIVON token to gain the right to be nominated for block creation. Validator has to stake 10,000 AIVON tokens in order to be eligible to start a node. For simplicity, we propose a fixed-sized

[3] https://jameshfisher.com/2018/01/09/how-to-hash-multiple-values.html
[4] https://ethresear.ch/t/initial-explorations-on-full-pos-proposal-mechanisms/925

deposits as multiple fixed-size validators can be ultimately controlled by the same entity. This amount has to be deposited in a single transaction using deposit() at root chain.

Validator's reward can be withdrawn without affecting the staked AIVON tokens or the state of the Validator node – does not cause it to go offline. Validators' rewards also does not count towards the stake increment.

Once a Validator is penalized due to misbehavior, Validator would no longer be considered active, due to having lower than the stake requirement. Validator's operator who wishes to continue to participate in Validator's pool would thus have to withdraw(), investigates and fixes the problems with the Validator node that causes the original misbehavior, and make a fresh deposit() of 10,000 AVO for stake.

7.3 AI & HE Nodes

AI Node's Proof Of Work

Assigned AI node is given a period of time to process a task. Task results in the form of metadata should be submitted to AIVON transaction pool as soon as it is completed.

As output of AI nodes are fully deterministic and identical, results are only fully accepted if at least 66% of all nodes provided the same results, i.e. 66% agreement. If the agreement is short of 66%, 2nd pass would be initiated, involving more AI nodes, by Validator. Once majority consensus is achieved with 66% agreement, the result of the majority will be accepted; those nodes whose result are different from the majority and not accepted will lose a significant portion of their collateral and have it redistributed to the nodes that submitted the accepted results.

It should be taken note that nodes which do not respond in time, i.e. after timeout period, should not be penalized as harshly as submitting non-acceptable result as we.

AI Node's Deposit

All AI nodes are required to stake a minimum deposit of 500 tokens to gain the chance to be selected to perform the AI task.

He Node's Proof Of Work

Assigned HE node are classified into 2 categories based on skills:

1. Low skilled task which can be gamified to produce a binary response from HE to help verify metadata result from Video Tagging type of Job. An example would be to ask HE to confirm if George Clooney is in the scene or if it is a scene in the park.
2. High skilled task which requires HE to moderate and edit the metadata previously generated by the AI node. These are required for transcription and translation type of Job.

For low skilled task, results are only fully accepted if at least 66% of all HE nodes provided the same results, i.e. 66% agreement. If the agreement is short of 66%,

another pass would be initiated, involving more HE nodes, by Validator. For high skilled task, it will follow the sequence below:

1. Assigned HE nodes will perform the task which will likely yield different results
2. Validator assign a group of HE review committee nodes to vote and rank the results of the 1st pass according to 3 buckets: “unacceptable”, “acceptable”, “best”. Results are accepted if at least 66% of the HE committee nodes vote for the same 1st pass result as the best.
3. If the agreement is short of 66%, another pass would be initiated, repeating (1) and (2) and increasing the HE nodes by x2+1.

Upon majority consensus, HE nodes which produce the “best” voted result would be awarded the reward. HE nodes with results that achieve at least 66% “unacceptable” voting would be penalized by forfeiting its stakes.

In the next phase of AIVON development, we intend to provide a separate training and evaluation service to help qualify the skillsets of the Human Experts. This can help level up the skillsets of Human Experts; depending on level of difficulty, qualified Human Experts will be awarded free AIVON tokens which could be used as staking deposit to increase their chances of winning HE tasks and earning more tokens. However, the design of this training service is beyond the scope of the technical paper.

He Node’s Deposit

All HE nodes are required to stake a minimum deposit of 50 tokens to gain the chance to be selected to perform the AI task.

7.4 Results Claim

Claimant of results can be done publicly or privately. Results claim are done by submitting a verifiable order of which the snippets that make up a full video.

Public Claim

If a Request submits a claim publicly, this effectively makes the generated metadata public, allowing AIVON Video Search Engine to index the video asset and generate the ContentGraph score. Anyone or other Dapps can also access to the blockchain knowledge of the video asset.

In order to encourage more public utilization of AIVON and enrich the content in AIVON blockchain, Requester that makes a public claim will be refunded a portion of the fee paid.

Private Claim

Should a Requester wish to use AIVON to generate metadata but does not wish to make the knowledge public, the Requester does not have to submit the claim to the blockchain.

Private claims are made outside of blockchain. For example, a Requester wishes to inform a publisher/platform of the processed metadata of a private video that has been processed by AIVON, the Requester simply has to submit the claim privately to the publisher/platform. The Publisher/platform would then be able to independently aggregate the processed metadata and generate the ContentGraph score from AIVON blockchain and also able to independently prove that the processed metadata is legitimate.

7.5 ContentGraph

ContentGraph is an aggregated metadata and vector of confidence scores of a video from the snippets scores that make up the full video.

ContentGraph is not stored on AIVON blockchain, only public claims are stored. ContentGraph to a video can be independently generated and verified by interested party from claims that are obtained, either publicly within AIVON blockchain or privately without blockchain.

8. Demand & Supply Of AIVON Token

8.1 Token Demand

Transaction Fee

1. Every transaction must pay transaction fee to validator as part of block reward.
2. Demand for AIVON token will increase as AIVON transaction volume increases.

Reward

1. Client must have AIVON token to request job execution.
2. Demand for AIVON token will increase as number and scale of job requests increase.

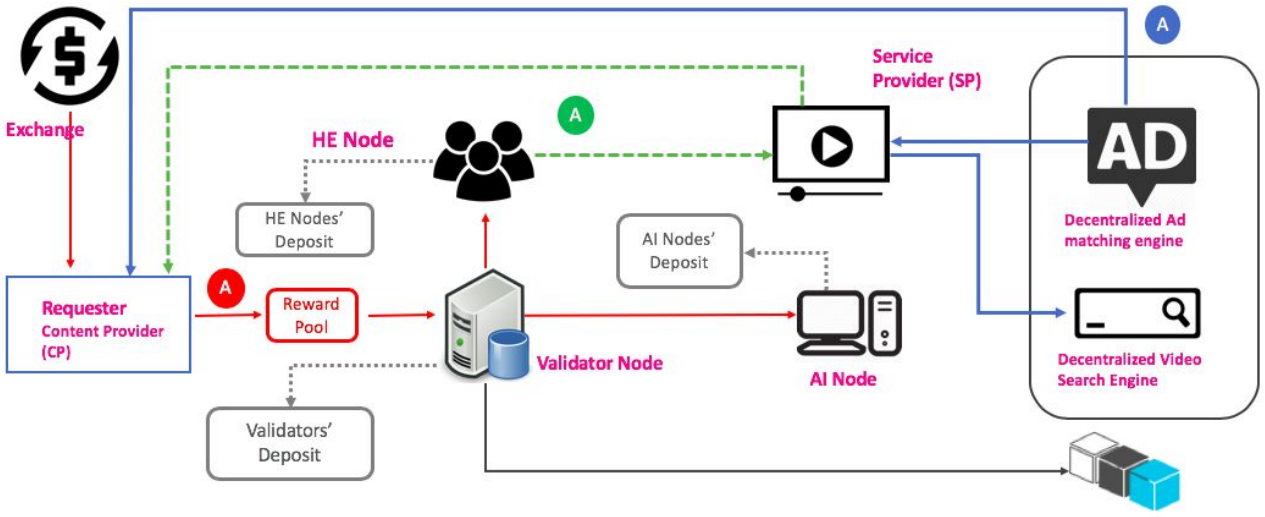
Deposit

1. All participants must put down deposit (Staking) in AIVON token to gain the right to participate and be selected.

2. Validator nodes must put down deposit in AIVON token to have the chance to be selected. AI/HE nodes must put down deposit in AIVON token to have the chance to be selected to perform the tasks.
3. Demand for AIVON token will increase as the number of nodes which want do jobs in AIVON increases.
4. Deposits through staking will reduce the amount of tokens in circulation.

Credit Tokens

1. To level up and qualify HE nodes, we envisage DApps that provide systematic online course to train and test humans on the high skilled tasks. (e.g. translation).
2. For humans who successfully complete the course and test, could be given free AIVON token credit which they could use to stake to participate but not to trade.
3. To enable this, the DApp owner could sign a message staking a UTXO with the stake amount authorizing a node's IP and a receiving address that is not controlled by the DApp owner for rewards.



8.2 Token Supply

Initial Distribution

1. 400,000,000 AIVON tokens will be created at Token Generation Event.

Mining Reward (Monetary Policy)

1. Total supply is capped at 1,000,000,000 tokens.
2. 600,000,000 additional AIVON tokens will be minted and distributed to Validator, AI/HE nodes as block rewards over the next 10 years. The block reward will decrease at a rate of 25% year-on-year over the 10 years.