SENNO

Decentralizing decision making

Senno whitepaper

DRAFT v0.9

“I have a dream for the web… and it has two parts… In the second part of the dream, collaborations extend to computers. Machines become capable of analysing all the data on the web – the content, links, and transaction between people and computers. A "Semantic Web" which should make this possible, has yet to emerge, but when it does, the day-to-day mechanisms of trade, bureaucracy and our daily lives will be handled by machines talking to machines. The intelligent “agents” people have touted for ages will finally materialize.”

Tim Berners-Lee, Weaving the Web, 1999

Senno is the Blockchain's first sentiment analysis platform with an open API for 3rd party apps.

It utilizes distributed sentiment analysis and advanced AI algorithms to create a Real-Time crowd wisdom ecosystem and sophisticated business intelligence analytics.

Senno will revolutionize the way decision making is made in the business and private sector.

Introduction

Over the last decade, Web 2.0 websites have become increasingly popular. Users have been interacting and collaborating with each other at an exponential rate. User-generated content is utilized for a wide range of applications, such as problem solving, news, entertainment, advertising, gossip and research. While many companies understand that they can't ignore user-generated content (like posts on social media) related to their brands or products, very few are considering this content as a means to understand consumer thinking, nor are they implementing crowd knowledge in their day-to-day business decisions.

The magic of user-generated content is in its unprecedented reflection of public opinion and life habits. Reading what others are writing can give a picture of their whole lifestyle, including where they live, what they look like, what they like to eat, where they hang out and more. This is because people involve others in their lives and aren't simply writing sterile statements with no context.

The mass sharing of user generated content allows us to analyze what consumers really think, rather than just getting answers to specific questions like in a survey. Understanding their way of thinking, what drives them to make a specific decision, how they express themselves and what terminology they use to describe a product leads to understanding how a financial asset, brand or product is perceived.

The influx of user-generated content

The development of the internet has created an immense and unprecedented space for publication. Anyone can share their opinions, whether through brief comments on Facebook or Twitter, or long, well thought-out diatribes on Wordpress or Medium.

Not only has the internet given individuals a space to express themselves, but corporations have also embraced the opportunity to interact with the public. From posting articles on LinkedIn to sharing whitepapers on their websites, the internet has given them a variety of new ways to communicate. On top of that, we also have the host of news and other media sites that have exploded in the new market that the internet has provided.

All of these digital spaces have led to a goldmine of sentiment data, which can be taken advantage of with increasingly sophisticated algorithms and the exponential growth of computing power. Now we have the tools and the information to truly understand public opinions. These developments mean that we can finally move past more primitive techniques such as educated guesses and estimations, which led to less-reliable results.

The need

The first sentiment analysis engines were complex, expensive, and were available only to a handful of entities. **Companies that have tried to find a solution have encountered 3 major issues:**

- It requires immense hardware resources to scan millions of web pages and analyze sentiments In real time. This has meant high operational costs and an expensive end product.
- A centralized solution that isn't transparent holds a risk of data manipulation which affects business credibility.
- In the early years of Web 2.0, user generated content was mainly confined to forums and produced by early adopters. Most people either didn't use the internet, or used it in a limited way, meaning that the voice of the crowd was not heard online.

Although various implementation of sentiment analysis technology have been released during the last decade, those high costs of development and servers infrastructure to hold vast amounts of data have led to a reality in which sentiment analysis based products are used mainly by big enterprises. According to McKinsey Global Institute a retailer utilizing big data to its full potential can increase its operating margin by more than 60%.[1]

SMBs are not the only ones which require a lower cost infrastructure for sentiment data. personal users, do not have tools to perform a specific research or survey based on sentiments and should they decide to apply crowd wisdom on their day to day decisions they are mostly limited to the size of their online social circles.

The solution

Senno is a decentralized, blockchain-based sentiment-analysis platform with an open SDK that uses distributed hardware. Senno was created to allow third parties to integrate sentiment and data conclusion tools into their own platforms (e.g Digital assistants, like Alexa and Siri, would be able to answer a whole new set of questions with high precision, based on real time user generated content like the feelings of the majority on a subject, a taste, whether something is funny or boring). Senno will enable companies and individuals to tap into and stay connected to the sentiment by getting real time indications of the public opinion on a specific entity in any field.

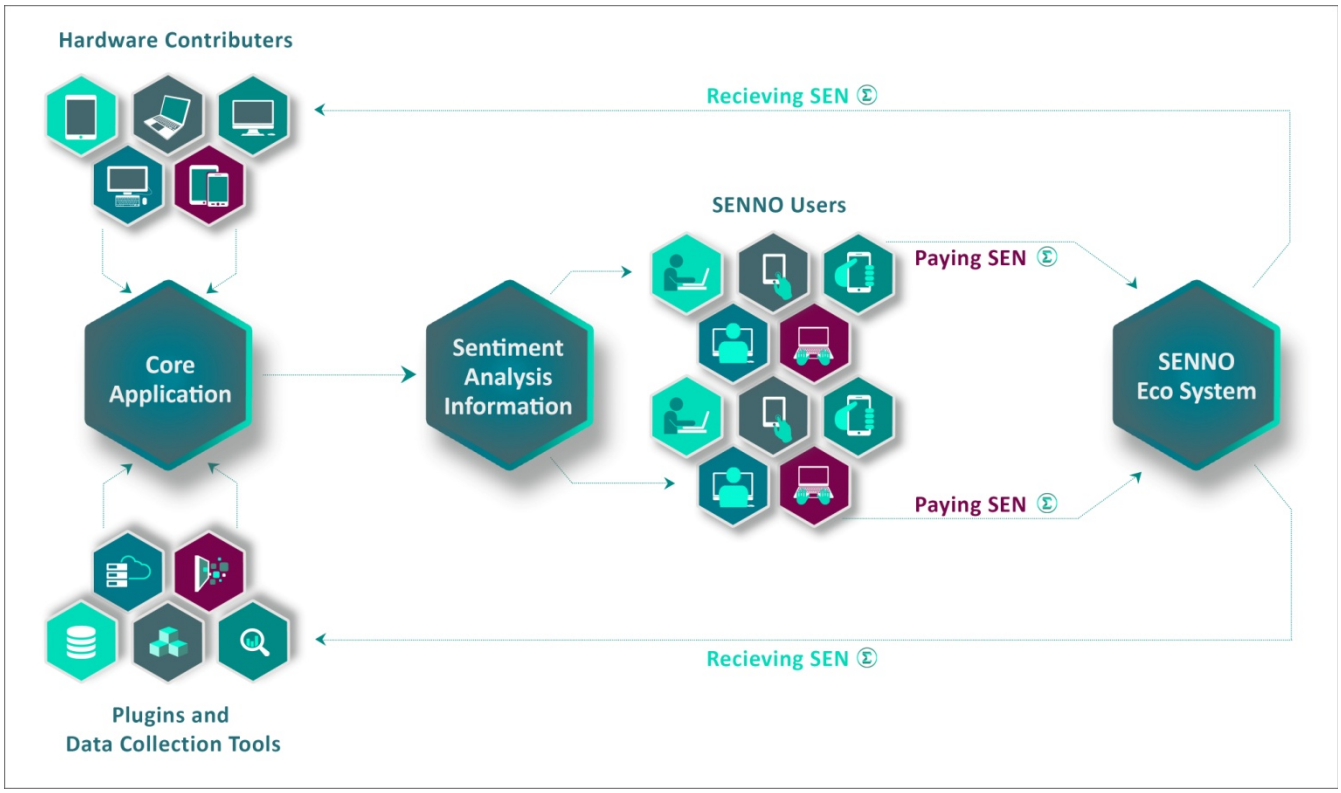
There are many benefits to tapping into public opinion in real time. These include being able to identify threats or opportunities and respond immediately, to monitor the impact of changes made to marketing campaigns and products as well as performance in a specific time frame or location. For example, an increase in the mentions of an entity like a company or a cryptocurrency may indicate that something is happening, but it is difficult to translate and act on these indications without verifying what they represent, and analyzing their meaning and strength.

[1] *Big Data The next frontier for innovation, competition, and productivity (PDF)*

Senno was designed to provide an infrastructure and SDK for third parties to accurately analyze the essence of user sentiments and to produce valuable information in real-time, which can then be responded to either manually or automatically.

With blockchain-based smart-contract technology, the engineers of Senno have come up with an innovative solution that delivers deep, precise, wide and real-time analysis of sentiment data to all the members of its community, all at a fraction of the cost of traditional solutions, with significantly higher accuracy.

Senno network usage flow



Senno Network usage flow

The key features of Senno

Contributions rewarding system[2]

30% of the SEN token supply will be locked in smart contracts and used for contribution rewarding.

- **Distributed Hardware for lowering costs**
  Senno uses the power of its blockchain network to allow members to contribute their hardware resources, which eliminates the need to maintain a costly data center and rewards the contributing members with SEN tokens.

- **Accessing private Data**
  Using blockchain. Senno users who provide access to their private data channels (IM app groups Telegram, WeChat, QQ, private news channels) will be rewarded by the system with SEN tokens. These private data streams are usually characterized by high quality sentiment value which will allow Senno to produce much more accurate analysis than what exists in the market today and based on publicly available data.

- **Development contribution**
  Community members will be able to earn SEN tokens by developing features and plugins for the system. The developed modules which will be voted into the system by the Senno community, will credit their developers with SEN according to the complexity and usage of the developed feature.

Flexible usage and accessibility

Senno has an open source, decentralized platform which enables integration by two methods:

- **Open SDK**
  A built in SDK that enables 3rd parties to easily develop applications and plugins that utilize Senno's core functionality. Applications developers would be able to integrate sentiment analysis capabilities into their developed software and provide alerts, signals and analytics regarding their area of expertise.

- **API connectivity**
  Senno will introduce an open API which will allow fast and dynamic data queries from Senno's distributed storage. Companies like McDonalds will be able to use the Senno API in order to pull sentiment data into their BI systems for measuring trends and sentiments in public opinion regarding new products and marketing campaigns.

[2] *See Hardware Contribution Rewarding section (p.22)*

Sentiment analysis AI

Using Artificial Neural Network (ANN), Senno's Machine learning algorithms will be able to identify high value data sources and grant them with higher internal scoring, thus providing more accuracy to the final sentiment evaluation, this mechanism will also optimize the amount of data required to produce the analysis.

Dynamic listeners

To obtain a high amount of diverse data, Senno has developed dynamic listeners, which monitor the classic web 2.0 channels (such as blogs, comments, YouTube channels and social media) as well as video and live broadcast transcriptions. The dynamic listeners can be directed to gather a specific type of information according to real time commands sent from the SennoCore. (See Appendix #2)

Accountability and tampering resistance

By using Blockchain technology, sentiment results are encrypted in transit then stored within a distributed Database and digitally signed to assure data integrity and revalidation which will provide full transparency and assurance that no manipulation has been occurred during the production of the sentiment result.

SennoApps

Senno will introduce an application marketplace which will include apps in multiple categories such as Financial, Medical, Marketing and more. This means that end users will be able to enjoy the benefits of sentiment analysis in almost every aspect of their lives, while developers would be able to introduce sentiment analysis based apps, for example: medical diagnostic apps, trading alerts, Business intelligence, crowd wisdom, marketing campaign effect measurements and more.

Digital Identity

Senno runs on the NEO blockchain to enforce data integrity and prevent manipulation. By using the NEO digital identity, Senno will be able to characterize trusted channels of data and assign them with a suited reputation score using its AI algorithms.

NEO – The Senno Blockchain

NEO smart economy is a blockchain-based project specifically designed to host smart contracts, ICOs and apps in a decentralized manner. It is open source, Turing complete and has a large community backing with professional and responsive teams. By using technologies such as P2P networking, dBFT consensus, digital certificates, superconducting transactions and cross-chain interoperability, the NEO blockchain enables management of smart assets in an efficient, safe and legally binding manner that surpasses all other blockchains.

Why Neo

Bitcoin and Ethereum have dealt with a range of security issues that have resulted in several forks of their respective blockchain platforms. This has led to a variety of workarounds to defend against spam and DDoS attacks, as well as to streamline their blockchains.

The proof-of-work algorithm requires vast amounts of energy, which limits how startups can use Ethereum's smart contracts. NEO uses a delegated Byzantine Fault Tolerance (dBFT)[3] system instead, which is more suitable for smart contracts. Although it compromises on availability, it ensures transaction finality and can support a greater number of transactions.

Transaction finality means that when a transaction has been confirmed, it is permanently recorded and cannot be revoked. We believe that this integrity is far more important than availability when it comes to financial applications. The dBFT mechanism has proved itself to be robust, because it hasn't suffered a single crash in the time it has been operating.

Additionally:

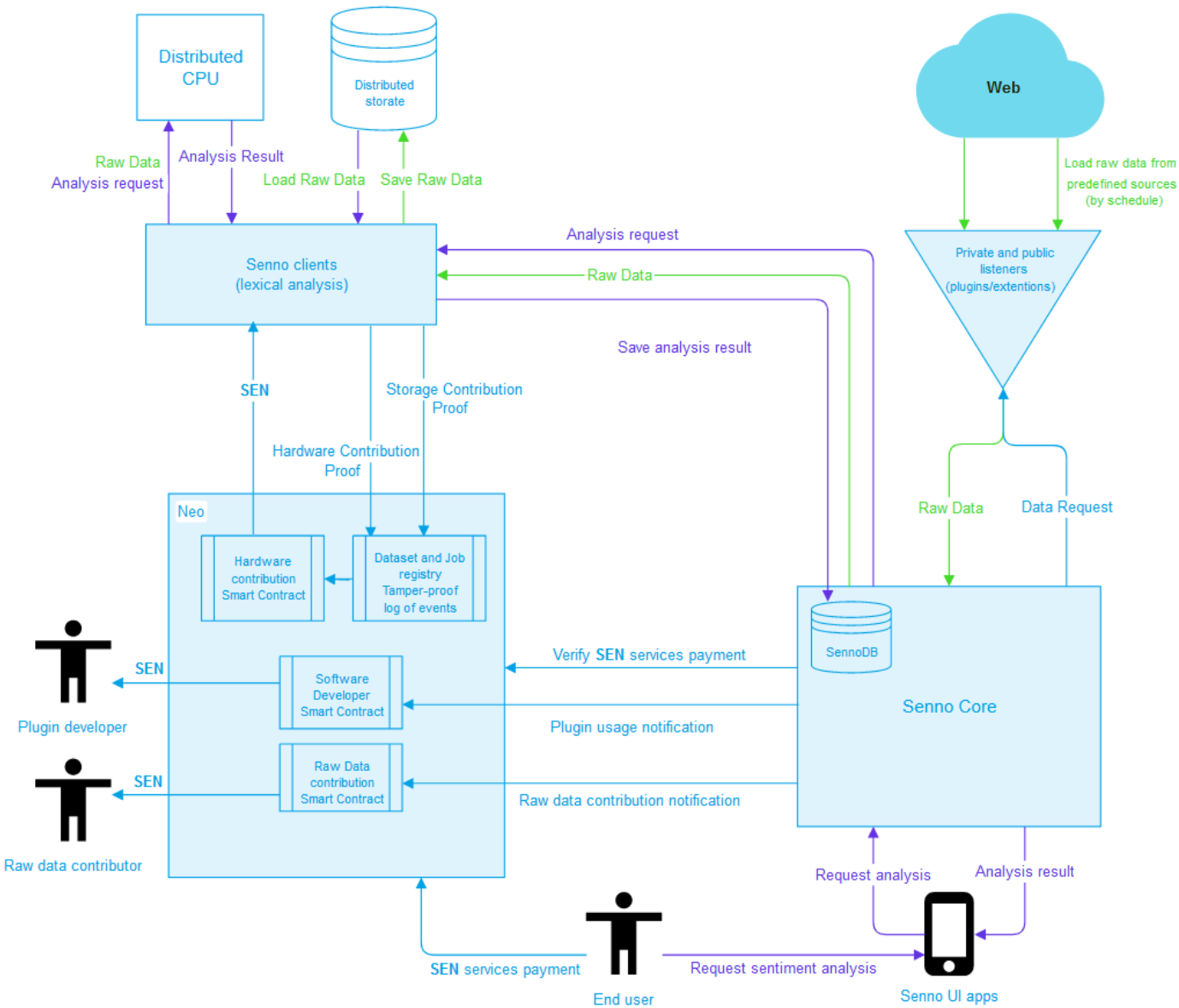
NEO's smart contracts and Apps can be written and compiled in **C#** and **Java**. In the future, developers will also be able to write smart contracts in **Python** and **Go**. This will drastically reduce the entry barrier for all developers around the world.
NEO is quantum computer-proof. Quantum computers are believed to have the ability to break into and hack the cryptographic math on which blockchains are based. NEO[7] indicates that they have already developed an anti-quantum cryptography mechanism called NeoQS.

- As for today, compared to Ethereum's 15 transactions per second, NEO supports up to 10,000 transactions per second, which is very energy-efficient compared with Ethereum.

[3] *Neo whitepaper (GitHub)*

Senno infrastructure overview

Senno was designed to efficiently popularize data stored on a decentralized environment and oversee the integrity of the information using blockchain smart contracts. The heart of Senno Network is the SennoCore which serves as a bridge between the off-chain world of non-blockchain applications and the on-chain smart contracts which are executed on the NEO blockchain.



Senno network architecture

The Senno Core

- Holds the business logic of the Senno network
- Handles the communication with external entities (SennoApps, API and SDK)
- Verifies balances and service payments with the blockchain.
- Distinguishes between queries for existing sentiments / new sentiment calculations.
- Manages the data listeners using A.I source prioritization. The raw data is divided into small chunks and saved into distributed data storage networks.
- Communicating with the SennoClient network to manage distributed hardware resources.

Data listeners

- Acting as a pluggable service architecture, allowing to add, change and update data providers.
- Enables retrieval of data from public and private channels.
- Collects raw data from designated channels according to dynamic prioritization from the Senno Core.

The Senno Client

- Retrieves processing and storage jobs from the Senno Core and executes them on its allocated resources.
- Validates contribution against the distributed hardware smart contracts on the blockchain.
- Constructs the sentiment analysis result and sends it to the core

NEO Blockchain

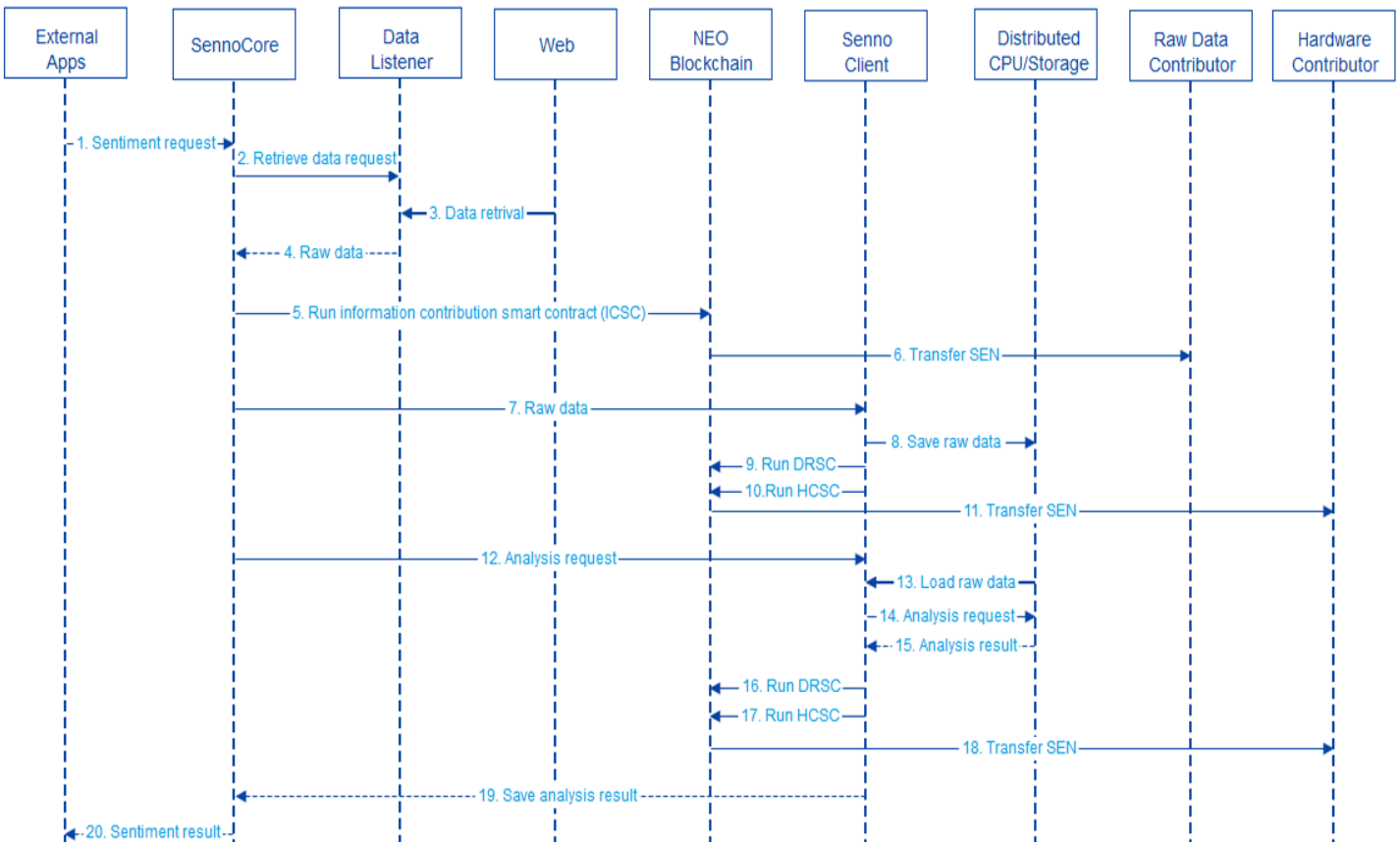
The on-chain implementation of Senno will be done by the following smart contracts (SC):

- Hardware contributor smart contract (HCSC). This SC will complete a transaction and pay reward to the hardware contributor once a verification of the storage contribution or lexical analysis has been made.
- Data registry smart contract (DRSC) validates the hardware contribution and calls HCSC.
- Software developer smart contract (SDSC). This SC will complete a transaction and will pay reward to the software developer once a data feed plugin has been used.
- Information contribution smart contract (ICSC). This SC will pay reward to Information contributors, which gave Senno listeners access to their private channel / group while keeping the data unrevealed.

Encryption

- Communication with external sources and 3rd parties will be encrypted using SSL/TLS.
- Raw data and analysis requests sent to clients will be encrypted with ECC asymmetric encryption.
- Data for distributed analysis will be divided in batches and mixed between different clients in order to avoid any possibility to track the source.
- Data in rest will be encrypted using AES algorithm.

Main flow sequence diagram



Senno main flow sequence diagram

Flow description:

1. A 3rd party sends a sentiment request to the SennoCore using either Senno's SDK or API.
2. Check if the sentiment request was produced in the past (as a result of a previous query), in case the result wasn't produced (does not exist in SennoDB), send the listeners a request to retrieve the specific new raw data.
3. The listener retrieves new raw data according to the dynamic parameters received from the SennoCore and encrypts it.
4. Listeners then send the new encrypted raw data back to the SennoCore.
5. If the data was received from a private channel, the Information contributor should be rewarded with SEN tokens by the information contribution smart contract (ICSC).
6. SEN tokens are transferred to the information contributor.
7. Encrypted raw data is sent to the Senno clients for processing.
8. The data is saved on the distributed storage.
9. The data registry smart contract (DRSC) is called to validate the stored data.
10. Once DRSC confirms the stored data, a payment is made to the storage contributor by the HCSC.
11. SEN tokens are transferred to the hardware contributor.
12. The sentiment request is sent from the SennoCore to the Senno client network.
13. The Senno client loads the raw data from the distributed storage to RAM.
14. Senno client sends the raw data and the analysis request to the distributed CPU.
15. The analysis result is calculated in the distributed CPU and the sentiment of the data is encrypted and returned to the Senno client.
16. The data registry smart contract (DRSC) is called to validate that the lexical analysis was calculated correctly.
17. Once DRSC confirms the calculation, a payment is made to the storage contributor by the HCSC.
18. SEN tokens are transferred to the hardware contributor.
19. Senno client sends the encrypted analysis result to the SennoCore which stores it in the SennoDB.
20. The SennoCore responds to the 3rd party with the sentiment result.

The math behind Senno

Lexical classification

When a domain lacks any labeled data, it is still possible to build models of sentiment classification with only background knowledge. An example is to use a lexicon that defines the polarity of certain words. This would assign terms as either positive or negative and they could then be measured based on their frequency of appearance in each text.

You can then compute the probability that an element *D* belongs to a positive class with $P(+|D) = \frac{a}{a+b}$ where *a* is the number of appearances of positive terms and *b* is the number of negative terms. A document will be deemed to belong to the positive class if $P(+ \mid D) > t$, where *t* signifies the threshold of classification. If $P(+ \mid D) < t$, the document will be deemed negative.

We assume that t=0.5 if there is no other information about the positivity and negativity of terms in the sample. This means that a document with more positive terms will be deemed positive (given that there is now prior information).

We use this approach as one of our baselines and call it the Lexical Classifier. The lexicon used by Senno was developed for other text mining purposes by the IBM India Research Labs. It includes 2,718 positive terms, 570 neutral terms and 4,911 negative terms that were labeled by the researchers. This lexicon[4] wasn't developed for any specific domain, which means that training examples should be used to understand the connotations in specific domains.

[4] *Lexical classification DB (CSV)*

Finding contributors

Ensuring that a sample of contributors[5] closely represents the crowd sentiment involves measuring how much the individual contributors and the entire sample-set agree with the majority opinion that prevails in the crowd. This is done using models that measure the quality of a contributor's estimate against the agreement of the crowd.

$L = \{I_1, I_2 .. I_m,\}$, $I_k : x \longrightarrow \{-1,1\}$ Denote the set of contributors and $\{X_1, X_2, \dots, X_t \dots, X_n)$ denote the sequence of examples, which arrive one at a time. We define $V_{it} = I_i (X_t)$ as I_i's vote on X_t and $S_t \subset \{ 1, \dots \dots , M\}$ as the set of contributors selected to contribute X_t. For each contributor I_i , we then define C_{it} as the number of times we have observed a contribution from I_i so far, and a_{it} as how many of those contributions were consistent with that set of contributors:

$$c_{it} := \sum_{\tilde{t}=1}^{t} \mathbb{1}_{[i \in S_{\tilde{t}}]}, \quad a_{it} := \sum_{\tilde{t}=1}^{t} \mathbb{1}_{\left[i \in S_{\tilde{t}}, V_{i\tilde{t}} = V_{S_{\tilde{t}}\tilde{t}}\right]}$$

Where $V_{St}t = \text{sign}(\sum I \in St\ VitQit)$ is the weighted majority vote of the contributors in S_t Contributor I_i's quality estimate is then defines as

$$Q_{it} = \frac{a_{it} + K}{c_{it} + 2K}$$

In this case, *t* is the number of examples that have been collected and labeled, while *K* is a smoothing parameter that yields a Bayesian shrinking estimate. This is the probability that contributor *I* agrees with the rest of the crowd. When there isn't enough data for more accurate estimation, values are pulled down toward ½. This means that those contributors who have seen fewer examples aren't given more weight than those who have seen more (these contributors are presumed to be able to judge with more certainty).

[5] *Channels which generate content related to our subject of the sentiment analysis*

Figure 1 – Crowd Sense algorithm's pseudo code (**C# implementation in appendix #1**)

1. **Input:** Examples $\{x_1, x_2, \ldots, x_N\}$, contributors $\{l_1, l_2, \ldots, l_M\}$, confidence threshold ε, smoothing parameter K.
2. **Define:** $L_Q = \{l^{(1)}, \ldots, l^{(M)}\}$, contributor id's in descending order of their quality estimates.
3. **Initialize:** $a_{i1} \leftarrow 0,\ c_{i1} \leftarrow 0$ for $i = 1, \ldots, M$.
4. **Loop for** $t = 1, ..., N$
   (a) Compute quality estimates $Q_{it} = \frac{a_{it}+K}{c_{it}+2K},\ i = 1, \ldots, M$. Update L_Q.
   (b) $S_t = \{l^{(1)}, l^{(2)}, k\}$, where k is randomly sampled from the set $\{l^{(3)}, \ldots l^{(M)}\}$.
   (c) **Loop for** $j = 3 \ldots M,\ j \neq k$
      i. $\text{Score}(S_t) = \sum_{i \in S_t} V_{it} Q_{it},\ \ l_{\text{candidate}} = l^{(j)}$.
      ii. If $\frac{|\text{Score}(S_t)| - Q_{l_{\text{candidate}},t}}{|S_t|+1} < \varepsilon$, then $S_t \leftarrow S_t \cup l_{\text{candidate}}$. Otherwise exit loop to stop adding new contributors to S_t.
   (d) Get the weighted majority vote of the contributors $V_{S_t t} = \text{sign}\left(\sum_{i \in S_t} V_{it} Q_{it}\right)$
   (e) $\forall i \in S_t$ where $V_{it} = V_{S_t t},\ \ a_{it} \leftarrow a_{it} + 1$
   (f) $\forall i \in S_t,\ \ c_{it} \leftarrow c_{it} + 1$
5. **End**

At the beginning of an online iteration to label new data, the contributor pool is initialized with 3 contributors; we select 2 contributors that have the highest quality estimates Q while one is chosen at random. These initial contributors allow the algorithm to achieve a balance between the exploration of the quality of the whole set of contributors and the exploitation of quality estimates.
Each contributor is asked to vote on the example, with a fixed price given for each label. The votes from the contributors generate a confidence score, which is calculated as:

$$\text{Score}(S_t) = \sum_{i \in S_t} V_{it} Q_{it}$$

a weighted majority vote from the contributors. Once we have this, we need to ascertain whether the Score(St) is an accurate representation of the vote from the majority of the crowd. If the result isn't clear we keep asking other contributors to vote until we have certainty on the result of the contribution. Our certainty is measured by looking at the value of |Score(St)|, then selecting whichever contributor outside of the St candidates has the highest quality Qit estimate. We use this contributor's vote to see if it would have changed the weighted majority vote if it had been included, or if it would have made the Score(St) close to zero.

If the Score(St) is close to zero the vote is approximately a tie, which brings us into the realm of uncertainty. The determination of whether to add a candidate contributor is made under the following formula:

$$\frac{|\text{Score}(S_t)| - Q_{l_{\text{candidate}},t}}{|S_t| + 1} < \varepsilon$$

In this formula, ε is the permitted level of uncertainty, $0 < \varepsilon \leq 1$. When (1) is true, the candidate contributor is added to become part of *St*, which gives us the contributor's vote for *Xt*. We recalculate Score(St) by using the same steps as above for whichever contributor outside of the *St* candidates has the next highest quality *Qit* estimate. If the candidate does not get added to *St*, the weighted majority vote is assigned as the predicted contribution in this example. We then go on to analyze the next example in the set.

Senno Output and provided information

The Senno Core can generate two highly accurate variables with the algorithm described above. It can also recommend analyses that takes the variables into account:

Buzz – How much discussion is occurring about the subject.

ΔB – This represents the amount of change in the standard deviation from the total amount of new mentions and the average for a specific time period.

Mood – The positive or negative sentiment value for a given subject.

ΔM – This represents the amount of change in the standard deviation from the calculated sentiment and the average for a specific time period.

Analysis recommendations:
We take both ΔB and ΔM into account when creating accurate real time analysis recommendations, making sure to have the lowest possible intervals between samples. Our algorithm will measure the change in buzz and the mood of each minute.

For example, we ran an analysis on the subject “Bitcoin” on 31 Mar 2017 when Japan’s Financial Services Agency (FSA) published an announcement at 12:00pm saying that Bitcoin will soon become an official currency in Japan. We expected to observe a huge positive Buzz regarding Bitcoin, so we looked at Senno’s output at 12:15pm. We measured the change in BTC Mood for the last 15 minutes in comparison to the average 15-minute intervals in the previous 24 hour period. We found the following data:

Current Buzz(B) for the last 15 minutes = 1M mentions

Avg B (measured in 5-min intervals over 24h) = 180K mentions

Avg ΔB (24h) = 20K comments

ΔB (B- (Avg B + AvgΔB)) = 800K comments

BTC token had a 5x buzz (B) change rising from max 200K avg (180K+20K std) references in 15 min to 1M in the last 15m (12:00-12:15). Out of the 1M references 700K are positive moods(P), 200K are negative moods(N) and 100K are neutral sentiments. Thus the signal will be:

$$Buzz(B) = \frac{\Delta B}{B} = 4 \Rightarrow \left(\frac{(1000000 - 200000)}{200000}\right) \text{X } 100\% = 400\% \text{ UP}$$

$$Mood(M) = \frac{(P - N)}{N} = 2.5 \text{ X } 100\% = 250\% \text{ UP}$$

Both indications of a Strong Positive Upward reference trend in a 15min timeframe. The behaviour of BTC price in these 15min correlates perfectly with the result as can be seen in the chart below:

BTC price on Mar 31, 2017 to Apr 1, 2017[6]

[6] *Graph and prices taken from Coinbase.com*

The Senno Token (SEN)

The Senno ecosystem is based on an open-source cryptographic token named SEN. SEN is the underlying cryptocurrency that drives Senno's semantic analytic platform. SEN is a NEP-5 token which makes it transferable and fungible. This consists of features that allow users to earn SEN and use it to pay for services within the ecosystem. They can earn tokens by doing active work (e.g plugin development), or passive work (e.g. sharing data from private channels)

Usage and Purpose

Activity in the Senno ecosystem is performed primarily using SEN, making the token an integral part of the Senno platform and the driver for its economy, Its uses include:

- Rewarding hardware contributors.
- Remuneration for software developers and providers of network resources.
- Payments to developers who provide sentiment tailored plugins that have been requested.
- Commissions for affiliates that send referrals.
- Fees for the subscription Senno's crowd wisdom data and API Usage

Deposit and Withdrawals

Senno will be accepting contributions in Bitcoin, Neo, Altcoins, credit card and wire transfers.

Senno will apply to be traded on all the major cryptocurrency exchanges.

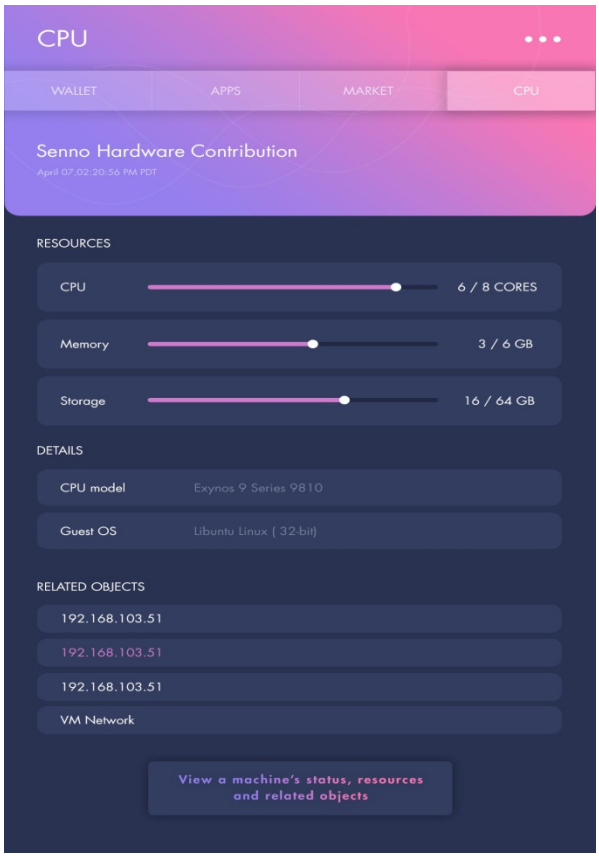
The SEN Token Contract

```
public class Contract : SmartContract
{
    public static object Main(string method, params object[] args)
    {
        string name = "Senno";
        string symbol = "SEN";
        BigInteger decimals = 0;

        var sencrptbts = new byte[] { 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1 };

        if (method == "deploy") return Deploy(sencrptbts);
        if (method == "totalSupply") return Storage.Get(Storage.CurrentContext, "supply");
        if (method == "name") return name;
        if (method == "symbol") return symbol;
        if (method == "decimals") return decimals;
        if (method == "balanceOf") return Storage.Get(Storage.CurrentContext, (byte[])args[0]);
        //Verify that the originator is honest.
        if (!Runtime.CheckWitness((byte[])args[0])) return false;
        if (method == "transfer") return Transfer((byte[])args[0], (byte[])args[1], BytesToInt((byte[])args[2]));

        return false;
    }

    private static bool Deploy(byte[] sencrptbts)
    {
        BigInteger initSupply = 10000000000;
        Storage.Put(Storage.CurrentContext, sencrptbts, IntToBytes(initSupply));
        Storage.Put(Storage.CurrentContext, "supply", IntToBytes(initSupply));
        return true;
    }

    private static bool Transfer(byte[] originator, byte[] to, BigInteger amount)
    {
        //Get the account value of the source and destination accounts.
        var originatorValue = Storage.Get(Storage.CurrentContext, originator);
        var targetValue = Storage.Get(Storage.CurrentContext, to);

        BigInteger nOriginatorValue = BytesToInt(originatorValue) - amount;
        BigInteger nTargetValue = BytesToInt(targetValue) + amount;

        //If the transaction is valid, proceed.
        if (nOriginatorValue >= 0 && amount >= 0)
        {
            Storage.Put(Storage.CurrentContext, originator, IntToBytes(nOriginatorValue));
            Storage.Put(Storage.CurrentContext, to, IntToBytes(nTargetValue));
            Runtime.Notify("Transfer Successful", originator, to, amount, Blockchain.GetHeight());
            return true;
        }
        return false;
    }
```

The Senno Client

Senno will introduce web and mobile applications which will provide access to its main features:

- Wallet for holding and transferring NEP-5 based SEN tokens.
- Management of hardware resource contribution to the Senno network.
- Applications market (SennoApps) will enable searching and downloading 3rd party applications which are based on Senno's SDK.



Senno Client

Recognizing Contribution

The Senno Network was designed to address the major barriers of expansion, monetization and scale in the market of financial content. We have created built-in contribution modules that reward people who provide meaningful and measurable contributions to the economy. This is based on our belief that the platform's expansion relies on a decentralized foundation as well as ongoing incentives for members of the community and developers.

Hardware Contribution Rewarding[7]

The core of the Senno ecosystem relies on the processing power contributed through the Senno clients that run on the machines of the community. 30% of the SEN token supply will be locked by the smart contracts and used for contribution rewards. Contribution rewarding is applicable by the following methods:

- Information contribution:
  Senno users which will provide access to their private data channels (IM app groups Telegram, WeChat, QQ, private news channels) will be rewarded by the network with SEN tokens based on the amount and quality of the raw data they contribute. Information contribution will be paid by the Information Contribution Smart Contract (ICSC).

- CPU contribution:
  For each H/s contributed to calculate a sentiment, contributors will be awarded with the relative amount of SEN based on the total amount of H/s contributed across the whole network in a given time frame.

- Storage contribution:
  For each KB contributed to store raw data, Once the data has been validated by the Data registry smart contract (DRSC), contributors will be awarded with SEN according to their relative part in the total amount of the DB contributed across the whole network.

  Hardware contributions are validated Onchain by the DRCS and paid by the Hardware contribution smart contract (HCSC), while the distributed processing and storage are made Offchain.

- Development contribution:
  Community members will be able to earn SEN tokens by developing features and plugins for the network. The developed modules will be voted into the system by the Senno community using the Senno Governance Plugin Approval System. Development which were voted in will credit their developers with SEN according to the complexity and usage of the developed feature.

[7] *Scheduled be released as part of the Senno Private network at Q3 19*

A Sustainable Economy

Cryptocurrencies need to drive sustainable economies in order to be successful in the long term. The value of the Senno ecosystem will grow as Senno is adopted by more users and providers, as well as when their activities on the network increase.

Live sentiment analysis will rely on users earning or acquiring SEN, with demand growing alongside the rate of adoption and in proportion to the number of hardware resource providers on the network. The Senno model provides a way for people who contribute resources to the network, plug-in creators and external developers to earn SEN for their contributions to the network.

Extensions for the platform

Senno was designed as a multi-tier platform to separate the business logic module (Senno Core, which performs the semantic analysis) from the extensions and plugins that are used to retrieve raw data from the web. This architectural structure enables the network's reach to expand faster and more easily by allowing any member to generate tailored extensions to the specific data source they see as relevant. Members that contribute new extensions that are voted into the network will be rewarded with SEN tokens.

Derived and customized versions

We encourage any organization or individual to change the official reference implementation so that they can release their own customized and branded client for the Senno network. An open-source customization SDK will be published alongside documentation and tutorials to help streamline the process. This will encourage other sentiment analysis companies to use the Senno network. Senno’s services can be used for a range of applications, including:

- Monitoring and periodically reporting on brand sentiment
- Mapping brand environments
- Mapping target audiences
- Researching trends
- Researching social data
- Custom segments research

There will also be documentation and support for a range of customization options. These include branding, colors and graphics; changes to the user interface; modifying the user experience for relevant segments (for industry-specific experiences); changing the sorting order and filters of display indicators for the process of analysis discovery; and setting addresses of providers for syndication fee settlements.

These customized and derived implementations will not be forks of the platform or the token, but simply different clients that provide different products or user experiences. All clients will all use the same network, with the same characteristics, and with sentiment analysis powered by the SEN token.

Affiliation and partnership

Senno rewards those who contribute to the growth of the network, with larger contributors receiving greater remuneration. Opinion leaders may be interested in receiving referral rewards for promoting Senno within their communities, or by granting Senno listeners access to their private channels. Marketing educational academies could receive SEN payments for each new member who joins the platform and subscribes to market information. Developers, hardware providers and end-users will also be encouraged to add modules to Senno's core or spread the word about Senno to earn additional SEN tokens.

Business model & Monetization

Senno keeps up with the world in different ways. First it scans social networks, news sites, blockchains, messaging applications and other data sources for information, then it extracts the relevant data according to predefined rules. Finally, it produces sentiment data streams which contain refined information and trends.

Senno is introducing the next level of business intelligence by aggregating and analyzing data across millions of digital sources in real time, then using machine learning to monitor and understand the internal and external threats, as well as the opportunities for businesses.

The retail market is expected to surpass $28 trillion in sales by 2019, so millions of potential clients will be seeking secure and high-quality business intelligence data . Senno can provide it at lower costs and with lightning-speed processing times.

The Senno network will generate two different revenue streams. The first is as a B2B platform for sentiment data streams and the second is as a B2C application maker:

Sentiment data streams – third party applications will be able to subscribe and execute analysis requests on different data sources. These will include both private (user contributed) and public sources. Prices will vary according to the source and number of requests and can start at as low as $100 per month. Thanks to the decentralized resources and personal contribution technology, Senno's pricing will be significantly lower than traditional sentiment analysis services. While companies like Semantria charge between $12,000 to $24,000 per year and sentiment analysis BI services such as BottleNose can cost as high as hundreds of thousands of dollars per year, Senno's yearly all-around solution will range from $1200 to $6000 only and will allow SMBs the opportunity to benefit from this innovative technology.

Crowd wisdom data applications – Senno will also develop its own crowd wisdom applications based on its sentiment analysis engine. These will illustrate the usage of the technology and expand the network. CryptoScanner is a great example of an application with mass market potential that provides significant value to any cryptocurrencies trader. Subscription fees will vary from $20 to $50 each month and the target market is estimated today at nearly 10M traders.

Bringing BlockChain and cryptocurrency to the masses

One of the main goals of the Senno network is to introduce cryptocurrency analysis and trading to the mainstream audience. At this stage cryptocurrencies and blockchain technology tend to only be popular with technology fans. There are still many people who do not know about cryptocurrencies or their potential, which creates a challenge in moving those who trade in traditional markets across to using cryptocurrency-based assets.

Cryptocurrencies are supported by blockchain networks that are accessible over the internet, so new-era market trends will be spread out and, in many cases, created in the online world. Traditional market traders and newbies to the financial world will have to learn how to handle immense volumes of information, the constantly growing number of financial analysis tools, the wide range of data sources, as well as how to follow the variety of promising cryptocurrencies that are being generated on a daily basis.

Senno was created to analyze it all and use the wisdom of the crowd to help its community to figure out which cryptocurrency they should trade and what will make them money. The powerful algorithms of Senno produce immediate sentiment analysis and trends while the decentralized peer-to-peer computer network power enables it to reach an accuracy rate that has never been seen before in traditional financial markets.

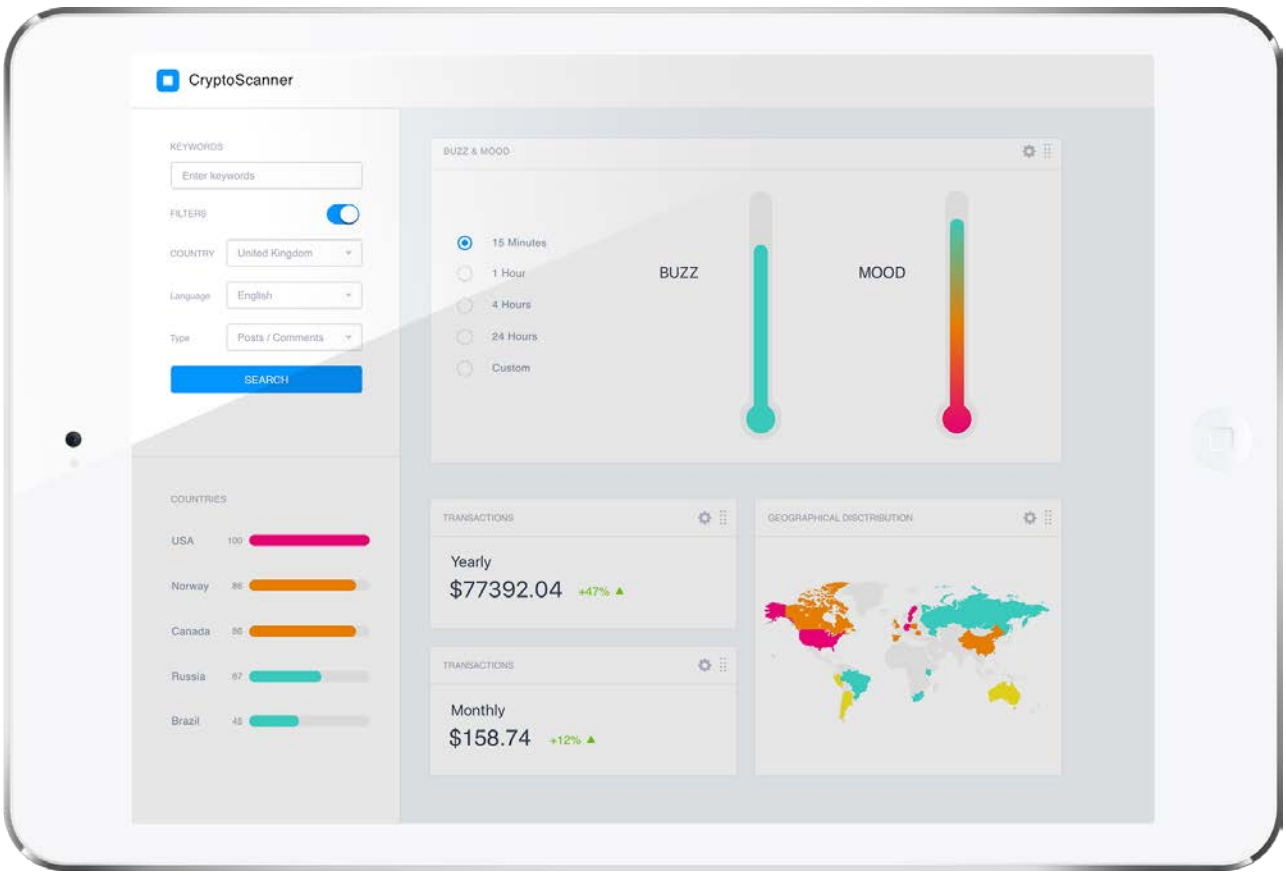
CryptoScanner - Senno’s First Application

Investors have been trying to beat the financial markets as long as they have existed. Over the years, people have developed a range of techniques to try and come out on top. These include analyzing data and looking for patterns, or plunging into the depths of a company's finances and operating structure to see whether it is undervalued.

Sentiment analysis is another key strategy. It involves getting inside the heads of the masses and determining how they feel about an asset. The aim to predict the changes and trade accordingly. This is a great technique in theory, but it requires accurate measurement and predictions of how people are feeling. Up until now, people haven't had much more than woeful forecasts and water cooler conversations to help guide their investments.

CryptoScanner Features

Senno will develop and publish a client for its platform in the form of a fully-functional, real-time, market-trading signal app for cryptocurrencies which shows accurate Buzz and Mood indicators for every coin at any chosen time frame. CryptoScanner will be open source and exhibit the implementation of the user-side module, including on-chain access over NEO to Senno smart contracts; and access to semantic content, reports and analysis from decentralized cloud storage.



CryptoScanner

Main Features:

- Real time scanning of crypto currency channels.
- Comparison of multiple crypto currencies.
- Filtering by Region, Country, Language and Type of Channel.
- Configurable alerts based on Mood and Buzz changes.
- API interface to leading exchanges to allow execution of trades.

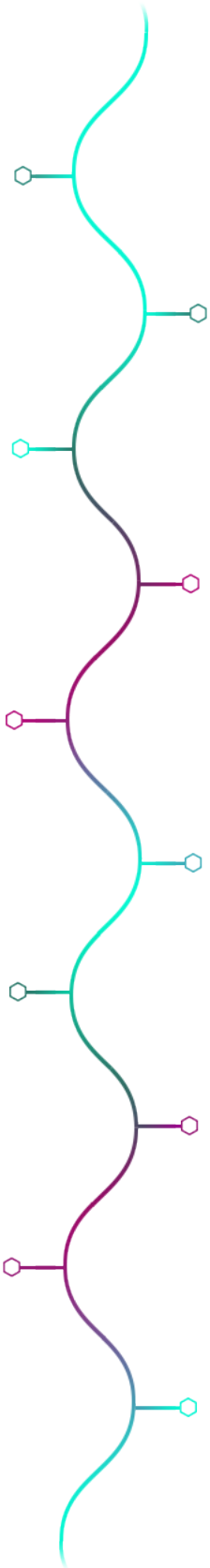
CryptoScanner will run on all popular operating systems - Windows, macOS, Linux, Android and iOS as well as an HTML5 web based client.

Reference Implementation

CryptoScanner will be published as open-source software on GitHub and serve as the formal reference implementation for a fully-compliant client to the Senno network. To maintain compatibility with the Senno network, the following aspects of the app must be maintained:

1) Reliance on SEN as the token of choice for all network activity
2) Usage of the official NEO smart contracts by the Senno platform
3) Compliance with the Senno P2P protocol for sharing hardware resources

ROADMAP



January 17
The birth of SENNO

April 17
First POC approval

November 17
Announcement

February 18
Website and
Whitepaper release

March 18
Crowd sale start

Q2 18
Fully developed frontend
offline sentiment analysis
neo smart contracts

Q3 18
Full MVP release
SEN token listing

Q4 18
CryptoScanner (Alpha)

Q1 19
CryptoScanner (Beta)
SDK and API release

Q2 19
SENNO Network launch

SENNO’s Token Sale Event

In order to finance Senno’s road map, Senno will conduct a token sale of 4B SEN tokens. The tokens will be sold with bonuses to the base price throughout the early stages of the sale. As of the conclusion of the sale, the distributed SEN will constitute the entirety of the available liquid supply. A portion of the supply would be pre-allocated to the Senno network, in a long term vesting schedule.

The token sale event will be divided into 3 stages with each offering a different level of discount:

- Private Sale
- **Pre Sale ($4M Cap)**
- Crowd Sale ($25M Cap)

Contribution methods: NEO, BTC, alt Coins, Credit card and Wire Transfer.

The pre sale date will be announced on 10th Mar 2018, **Pre sale will continue for 45 days**, or until the cap of $4M is reached. Participation instructions may be updated throughout the different phases of the sale and published on the Senno.io website.

Token Sale economy

1) Soft Cap: $2M
2) Hard Cap: $25M
3) Total Supply: 10B SEN
4) Sale Stages:

A. Seed ("Pre-Sale")

Base Price: $0.01 = 1.6 SEN

Contribution	Bonus	Bonus Vesting	SEN/cent
500-15K	20%	1 Month	1.92
15K-30K	25%	1 Month	2.00
30K-90K	30%	1 Month	2.08
90K-250K	35%	1.5 Months	2.16
250K-500K	40%	2 Months	2.24
500K-1M	45%	2.5 Months	2.32
1M+	50%	3 Months	2.40

B. Crowd Sale("ICO")

Base Price: $0.01 = 1.6 SEN

Contribution	up to $50k	$50 - $100k	$100k+
First 48 hours	10%	15%	20%
Week 1	5%	8%	10%
Week 2	3%	5%	7%
Week 3	1%	2%	3%
Week 4	0%	0%	0%

5) Remaining public tokens which are not sold will be burned.

6) Token vesting period:
   - Bonus vesting applies only on the Pre sale bonuses. Vesting period will start when SEN is listed in the exchange.
   - Partners and advisors: 24 months vesting, 6 months cliff tokens released proportionally every 3 months.
   - Employees – 36 months vesting, 1 year cliff, tokens released proportionally every 3 months.

7) Lock transfer for 30 days after sale end.

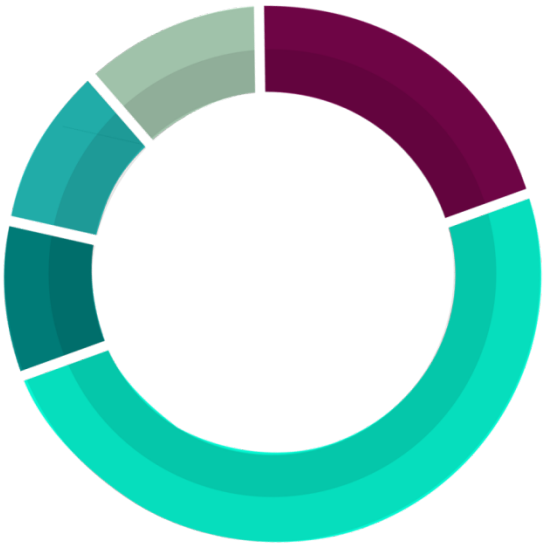
Use of Funds

Projected use of company funds

10%
Legal and accounting expenses

10%
Operational expenses: admin / HR / other staff, rent, utilities.

10%
Management expenses



20%
Marketing & sales staff, travel expenses

50%
R&D, blockchain and NLP experts

SEN Token Distribution

15%
Company shareholders (Excluding employees)

5%
Company employees

10%
Advisors and Business partners

30%
Reward tokens held by SENNO's smart contracts

40%
Crowdsale with funds allocated to the company

Our Team

ELAD PELED
Founder

BlockChain savvy and a serial entrepreneur, Founder of LibraTrade™ Inc, A financial technologies company which developed the world's first Smartphone trading platform in 2008. Served at the elite IDF intelligence unit where he mastered his professional skills.

RUDY ZAKUTO
Co-Founder and CTO

Technology and BlockChain Expert with a deep understanding of the BlockChain architecture, development of smart contracts for NEO and consensus algorithms. Over 20 years of hands on development experience with Specialization in online marketing.

LENNON TAM
COO Asia Pacific

Fintech professional and a BlockChain industry leader in Asia Pacific Market, Over 7 years of experience as a wealth manager in the banking industry, Mr Tam is also a certified financial planner and a member of the million dollar round table trade association.

CARLOS MANTILLA GUEVARA
BlockChain Developer

KIRILL LUKIANENKO
Senior Developer

TATYANA ARKHIPOVA
Senior Developer

NOA OMER
Graphic Design

CHENXI ZHANG
Client side Developer

CLAUDIA CASTANEDA
PR Manager

CHARLES C FLY
Server side Developer

Advisory Board

OFIR GARTNER
Founder STOX

MARC KENIGSBERG
Founder BlockSmarter

MICHAEL GREENBERG
Founder Financial magnets

VESEVOLOD PELIPA
CEO at SolarLab

ALEXANDER CHALY
Tech Advisor

ALEXANDER VOLKOV
Tech Advisor

VAN YUEN
Financial Advisor

GARY BERNSTEIN
BlockChain advisor

Our Partners

Risk factors

Possession and use of the token is a vote of confidence in the success of the platform, as well as a means of early support. That being said, there are several risks that all participants in the token sale should be aware of.

The following are the risk factors in relation to Senno' business and the Token Sale event in particular:

- Senno is a complex software platform and its launch may be significantly delayed due to unforeseen development barriers.
- Competition may introduce different sentiment analysis solutions and cause Senno to lose market share and eventually fail to deliver on its business goals.
- International laws and regulations may render the SEN trade impossible.
- The use of SEN tokens may come under the scrutiny of governmental institutions.
- The ownership of SEN tokens may fall under new and unpredictable taxation laws that will erode SEN benefits.
- Senno may not succeed in creating the necessary momentum and acceptance for the SEN token which may result in low liquidity and depletion of trades.
- The positions and plans outlined in this white paper may be altered as the project progresses.

Regulatory strategy

As a sentiment analysis platform, Senno is designed with decentralized infrastructure for applications in various industries and markets. As such, a large set of activities may be subject to regulatory scrutiny in various territories. This large and varied set may need to comply with regulation for securities trading, for financial institutions and for money services businesses, as well as for data copyright and for user privacy. Complying with each of these regulations, and sometimes in each jurisdiction, requires experience and comes at a great cost.

In decentralized applications, no single entity has control over the infrastructure required for its operation. This allows separating the infrastructure (sentiment analysis) from the app content. The resulting structure has the advantage of modular isolation: separate functions in the system can operate in isolation from one another, and possibly be provided by different vendors.

Modular isolation brings an advantage to companies and individuals when it comes to regulatory compliance: each only needs to comply with regulation in a limited field, lowering the cost of creating the required expertise and making it easier for new businesses to start up.

References

[1] S¸ eyda Ertekin, Haym Hirsh, Cynthia Rud in MIT Sloan School of Management, Massachusetts Institute of Technology Department of Computer Science, Rutgers University, *Approximating the Wisdom of the Crowd,2015, pp.2-6*
http://web.mit.edu/seyda/www/Papers/NIPS11Workshop_ApproximatingWC.pdf

[2] A. J. Quinn and B. B. Bederson, *"Human computation: a survey and taxonomy of a growing field," in Proceedings of the 2011 Annual Conference on Human Factors in Computing Systems (CHI), 2011, pp.1403–1412.*

[3] Bo Pang and Lillian Lee*. "Opinion mining and sentiment analysis"Foundations and Trends in Information Retrieval Vol. 2, No 1-2 (2008) 1–135*
http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.147.2755&rep=rep1&type=pdf

[4] G. Ramakrishnan, A. Jadhav, A. Joshi, S. Chakrabarti, and P. Bhattacharyya. "*Question answering via Bayesian inference on lexical relations. In ACL Workshop on Multilingual Summarization and Question Answering"*, 2003 *https://aclanthology.info/pdf/W/W03/W03-1201.pdf*

[5] Bing Liu. *"Sentiment Analysis and Opinion Mining",* Morgan & Claypool Publishers, May 2012. https://www.cs.uic.edu/~liub/FBS/SentimentAnalysis-and-OpinionMining.pdf

[6] Federico Alberto Pozzi, Elisabetta Fersini, Enza Messina, Bing Liu "*SentimentAnalysis in Social Networks*" 2017

Appendix #1 Crowd Sense implementation

```
using System;
using System.Collections.Generic;
using System.Linq;
namespace Main
{
    class CrowdSense
    {
        public class Example { }
        public class Labeler
        {
            public double[] QualityEstimate { get; set; }
            public int[] NumberOfConsistentContributions { get; set; }
            public int[] NumberOfContributions { get; set; }
            public int[] Votes { get; set; }
        }
        public void Calculate(List<Example> examples, List<Labeler> labelers, double threshold, double smoothingParameter)
        {
            Random rnd = new Random();
            var n = examples.Count;
            var m = labelers.Count;
            Init(labelers, n);
            for (var exampleIndex = 0; exampleIndex < n; exampleIndex++)
            {
                CalculateQualityEstimate(labelers, smoothingParameter, exampleIndex);

                var selectedLabelers = GetSelectedLabelers(labelers, threshold, exampleIndex, rnd, m);

                UpdateNumberOfContributions(selectedLabelers, exampleIndex);
            }
        }

        public static void Init(List<Labeler> labelers, int n)
        {
            foreach (var labeler in labelers)
            {
                labeler.QualityEstimate = new double[n];
                labeler.NumberOfContributions = new int[n];
                labeler.NumberOfConsistentContributions = new int[n];
            }
        }
        public static void CalculateQualityEstimate(List<Labeler> labelers, double smoothingParameter, int i)
        {
            foreach (var labeler in labelers)
            {
                labeler.QualityEstimate[i] = (labeler.NumberOfConsistentContributions[i] + smoothingParameter) /
                                             (labeler.NumberOfContributions[i] + 2 * smoothingParameter);
            }
        }
        public static List<Labeler> GetSelectedLabelers(List<Labeler> labelers, double threshold, int i, Random rnd, int m)
        {
            labelers.Sort((x, y) => x.QualityEstimate[i].CompareTo(y.QualityEstimate[i]));
            var randomLabelerIndex = rnd.Next(2, m);
            var selectedLabelers = new List<Labeler> { labelers[0], labelers[1], labelers[randomLabelerIndex] };
            for (var j = 2; j < m; j++)
            {
                if (j == randomLabelerIndex) continue;
                var score = selectedLabelers.Sum(sc => sc.QualityEstimate[i] * sc.Votes[i]);

                var current = (Math.Abs(score) - labelers[j].QualityEstimate[i]) / (selectedLabelers.Count + 1);
                if (current < threshold)
                {
                    selectedLabelers.Add(labelers[j]);
                }
                else
                {
                    break;
                }
            }
            return selectedLabelers;
        }
        public static void UpdateNumberOfContributions(List<Labeler> selectedLabelers, int exampleIndex)
        {
            var sign = Math.Sign(selectedLabelers.Sum(sc => sc.QualityEstimate[exampleIndex] * sc.Votes[exampleIndex]));
            foreach (var sel in selectedLabelers)
            {
                sel.NumberOfContributions[exampleIndex] += 1;
                sel.NumberOfConsistentContributions[exampleIndex] += (Math.Sign(sel.Votes[exampleIndex]) == sign ? 1 : 0);
            }
        }
    }
}
```

Appendix #2 Data Listener Interfaces

```
using System;
using System.Collections.Generic;
namespace SocialNetwork
{
    public class BaseEntry
    {
        public string ID { get; set; }
        public DateTime Date { get; set; }
        public string Text { get; set; }
    }
    public interface IFacebookExtractor
    {
        List<BaseEntry> GetPageItems(string pageId, int count);
        List<BaseEntry> GetComments(string objectId, int count);
        List<BaseEntry> GetMentions(string pageId, int count);
    }
    public interface ITwitterExtractor
    {
        List<BaseEntry> GetUserTweets(int userId, int sinceId, int maxId, int count);
        List<BaseEntry> GetHashtagTweets(string hashtag, int sinceId, int maxId, int count);
        List<BaseEntry> GetMentionTweets(int userId, int sinceId, int maxId, int count);
        List<BaseEntry> GetListTweets(int listId, int sinceId, int maxId, int count);
    }
    public interface IGooglePlusExtractor
    {
        List<BaseEntry> GetUserActivities(string userId, int count);
        List<BaseEntry> GetActivityComments(string activityId, int count);
    }
    public interface IInstagramExtractor
    {
        List<BaseEntry> GetUserMediaCaption(int userId, int minId, int maxId, int count);
        List<BaseEntry> GetHashtagMediaCaption(string tagName, int minTagId, int maxTagId, int count);
        List<BaseEntry> GetMediaComments(int mediaId);
    }
    public interface ILinkedInExtractor
    {
        List<BaseEntry> GetUserShares(string ownerId, int offset, int count);
        List<BaseEntry> GetUserArticles(string ownerId, int offset, int count);
        List<BaseEntry> GetComments(string ownerId, string commentId, int offset, int count);
    }
    public interface IRedditExtractor
    {
        List<BaseEntry> SearchSubreddits(string subreddit, string query, int count);
        List<BaseEntry> GetComments(string subreddit, string link, int count);
    }
    public interface ITumblrExtractor
    {
        List<BaseEntry> GetBlogPosts(int blogId, int offset, int count);
    }
    public interface IVkExtractor
    {
        List<BaseEntry> GetUserPosts(int ownerId, int offset, int count);
        List<BaseEntry> GetComments(int ownerId, int postId, int offset, int count);
    }
    public interface ITelegramExtractor
    {
        List<BaseEntry> GetHistory(int chatId, int offset, int count);
    }
}
```

New Trend Centre 704/811
Prince Edward Road East
San Po Kong
Hong Kong

info@Senno.io