TrakInvest AI Engine: Unlocking Market Insights from Heterogeneous Social Data

Bobby Bhatia
Founder and CEO, TrakInvest

Partha Talukdar
Founder and CEO, Kenome
Assistant Professor, IISc, Bangalore

December 14, 2017

Proliferation of the Internet and the World Wide Web have resulted in the explosive growth of online prediction communities and rich social investing datasets, examples include financial news, forum discussions, social media feeds, telegram channels, and many more. Such datasets hold the promise of providing accurate measures of investor trust for the always-connected modern investor looking for credible trade ideas and timely advice. Through these datasets, it is possible to evaluate investors not just based on their popularity, but also by their historic performance and recommendations.

In spite of the promise, a few challenges remain in ready utilization of such data sources. Firstly, there are numerous such datasets scattered across many different sources. Secondly, these data sources vastly differ in their structure and organization. For example, while a dataset containing historical stock prices is structured, news and social media feeds are inherently natural language-based unstructured data sources. Thirdly, data is generated through these sources at a very fast rate. For instance, tens of thousands of messages are posted per day in a single telegram channel. Fourthly, quality and trustworthiness of information can vary drastically from one source to another. These problems are even more acute in rapidly involving investment opportunities, such as cryptocurrencies. In short, the diversity, volume, speed, and veracity related challenges outlined above makes it humanely impossible for a retail investor to derive useful trading insights out of these heterogeneous datasets in a timely manner.

TrakInvest's AI Engine overcomes this challenge by creating the TrakInvest Knowledge Graph (TKG) – a curated and unified view over all the heterogeneous datasets of interest. The three key initiatives in the TrakInvest AI engine include:

- TrakInvest Knowledge Graph (TKG)
- Sentiment Engine
- Continuous Learning

A schematic overview of the TrakInvest AI Engine is shown in Figure 1. In the sections below, we present brief overview of each of the three components, more detailed technical methodology is available on request.

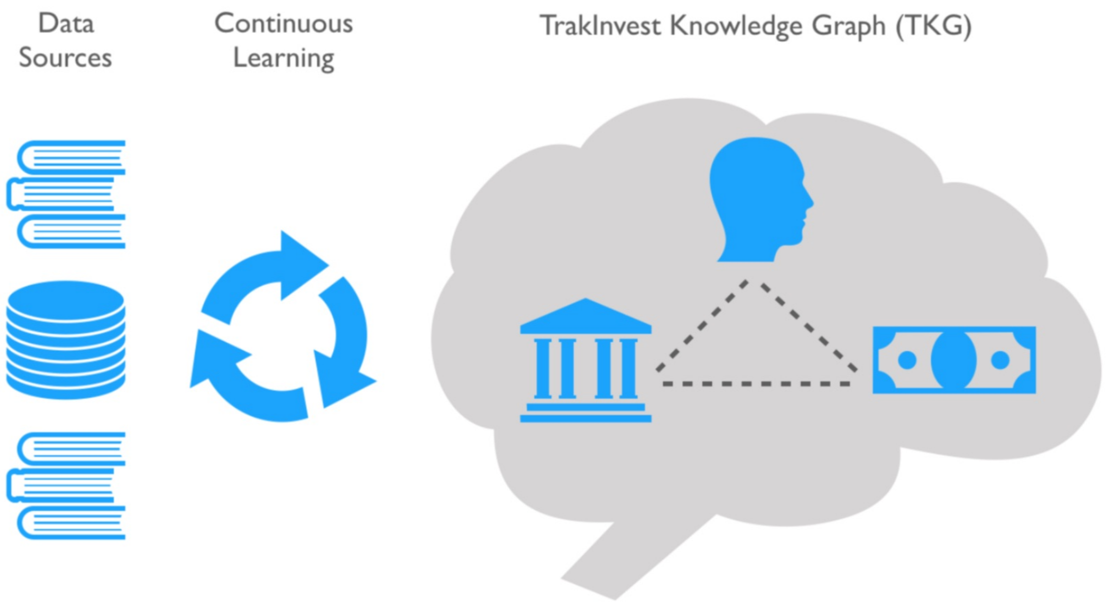

Figure 1: Schematic overview of the TrakInvest AI Engine. The engine first builds the TrakInvest Knowledge Graph (TKG) by automatically reading all the structured and unstructured datasets of interest. The engine also estimates crowd sentiments over various components of the TKG, e.g., investor sentiments over securities, companies, etc. The AI engine updates structure of the TKG, sentiment over its components, and also estimates predictive power of such sentiments against actual market indices using a continuous learning mechanism.

TrakInvest Knowledge Graph (TKG)

Knowledge Graphs (KGs) are multi-relational graphs consisting of entities and relationships among them. Knowledge Graphs have emerged as an effective way to extract and organize knowledge from large unstructured datasets. Leading search engines, such as Google and Bing, use such KGs to improve web search experience. For example, the knowledge panel on the right side of the search results page of such search engines are enabled by an underlying knowledge graph [1]. Freebase and DBPedia are examples of KGs constructed using manual contribution, while Yago, another popular KG, is built out of semi-structured data sources. Over the last few years, several techniques to build Knowledge Graphs (KGs) from large unstructured text corpus have been proposed, examples include NELL [Mitchell et al., 2015] and Google Knowledge Vault [Dong et al., 2014]. Such KGs consist of millions of entities (e.g., *Oslo*, *Norway*, etc.), their types (e.g., *isA(Oslo, City)*, *isA(Norway, Country)*), and relationships among them (e.g., *cityLocatedInCountry(Oslo, Norway)*). In spite of this progress, a KG designed exclusively to satisfy trading related information need has been missing.

TrakInvest's AI engine fills this important gap and leverages recent advances in automatic KG construction to build the TrakInvest Knowledge Graph (TKG). The TKG construction engine continuously scans a set of news sites, corporate press releases, social media feeds, and online discussions forums to build the TKG. The TKG consists of entities such as investors, analysts, institutions, securities, and cryptocurrencies; and relations among them such as *recommendedBy(analyst, security)*. Crawlers are written to derive information from the online sources mentioned above in a time-sensitive manner. This construction process is further supplemented by information mined internally from TIs discussion boards to provide an asymmetric advantage. The engine also extracts and maintains over time the relations among these different types of entities, along with the strength of such relations. A snapshot of the TKG is shown in Figure 1. We present below a few of the subproblems TrakInvest's AI Engine solves to build the TKG.

- **Predicate Schema Induction**: The KG construction techniques mentioned above are ontology-guided, as they require as input list of relations, their schemas (i.e., their type signatures, e.g., *recommendedBy(Analyst, Security)*), and seed instances of each such relation. Predicates and their schemas are the building blocks of any KG. Together they define the Ontology of the KG. Identification of predicates and their schemas is the first step towards building a domain-specific KG. Given unstructured data from a given domain, it is often not clear what predicates are necessary to capture the knowledge from that domain. Motivated by this need, recent research has focused on the problem of inducing binary relation (predicate) schemas from domain-specific data [Nimishakavi et al., 2016]. Following this line of work, TrakInvest's AI engine solves a joint tensor-matrix factorization objective to induce schemas of predicates from unstructured natural language data.

- **Instance Population**: Once the predicates have been identified, in the next phase, TrakInvest's AI Engine aims to identify instances of these induced predicates. In order to achieve this goal, the engine uses coupled multi-view semi-supervised learning [Mitchell et al., 2015]. The coupling constraints are automatically learned by performing inference over the already constructed KG. Instances for each predicate are learned by multiple machine learning algorithms, viz., Maximum Entropy, Recurrent Neural Networks (RNN), Conditional Random Fields (CRFs) etc. Predictions from each of these algorithms are aggregated together using a constraint satisfaction problem to build the final TKG. We note that TKG construction is an iterative process, where current KG is used to retrain the ML models, which in turn help build the next version of the TKG.

- **Overcoming Sparsity using Deep Reinforcement Learning**: In order to overcome sparsity in automatically constructed KGs, entity-centric densification techniques have received special attention in the recent past [Hegde and Talukdar, 2015, Sharma et al., 2017]. Building on this line of research, TrakInvest's AI Engine poses the KG densification problem as learning a policy over a Markov Decision Process (MDP), where the long-term cumulative reward obtained by taking action a from state s is given by $Q(s, a)$. Optimal Q-value is estimated using the Bellman equation presented below.

  $$Q_{i+1}(s, a) = E_{(s,a)}[R(s, a) + \gamma \max_{a'} Q_i(s', a')|s, a]$$

  For high dimensional state spaces, Deep Q Network (DQN) [Mnih et al., 2015] approximates $Q(s, a)$ as $Q(s, a, \theta)$ using parameters θ of a Deep Network.

- **Source Credibility Estimation**: Given the large number of data sources from which TrakInvest's AI Engine extracts knowledge from, not all sources involved are likely to be equally credible. Moreover, credibility of sources may vary in a topic-dependent manner. TrakInvest's AI Engine poses source credibility estimation as a semi-supervised learning problem over graphs using Probabilistic Soft Logic [Samadi et al., 2016].

- **Natural Language KG Inference**: Successful resolution of the subproblems mentioned above results in the construction of the TKG. In order to make sure individual investors are able to effectively interact with the TKG to gain valuable trading insights, TrakInvest's AI Engine also offers convenient Natural Language Q&A interfaces to the TKG. In order to accomplish this goal, the engine leverages KG inference techniques such as [Gardner et al., 2013, Gardner et al., 2014].

Sentiment Engine

TrakInvests sentiment engine reads sentiments from unstructured text data and aggregates such sentiments over entities and relationships in the TKG. For example, this sentiment engine will extract and store in the TKG how an analysts recommendation for a particular security has evolved over time. Similarly, this engine will estimate how crowd sentiment over a particular cryptocurrency has changed over time. The results of this sentiment analysis will work in both a push and pull mode. In push mode, news and announcements will trigger a push notification to end users based on their interest profile while pull actions will empower the end user to query the engine on the current status of a current stock. Once the textual information has been retrieved, it is then preprocessed for vectorisation. A multi-pronged approach is then taken to determine sentiment based on consensus readability measure (e.g., Fog Score), lexicon based approaches (bag-of-words using Harvard General Inquirer / Loghran and McDonald dictionaries) and machine learning methods including Support Vector machines, Random Forests and Deep Neural Networks, trained over historical price data of each share of interest. Thus, the TrakInvest Knowledge Graph, annotated with sentiment over its various components, provides a holistic overview of the market condition covering various stakeholders as mentioned above. In light of ever-changing world conditions, this graph is continuously updated and expanded as necessary.

Continuous Learning

The recommendations from the TKG and sentiment analysis engine will then be supplemented with a TI-STOCK score, a TI-ANALYST score, and a TI-TRADER score, each providing a time sensitive quantitative metric by which the stocks, analysts, and traders can be ranked. These scores are computed and updated using a continuous learning mechanism. The scores are first appropriately parameterized, with the parameters updated by correlating the predictive power of the corresponding variable against real market signals. Let us consider the analyst scores as an example. Through this learning mechanism, it will be possible not only to estimate overall competency of an analyst, but also learn about her biases in a data-driven manner. This goes far beyond the conventional popularity and perception-based evaluation of an analyst, providing the modern investor with an unbiased ranking of analysts. The continuous learning mechanism uses state-of-art Network Analysis, Deep Learning over Graphs, and Graph Embedding techniques to estimate these scores.

References

[Dong et al., 2014] Dong, X., Gabrilovich, E., Heitz, G., Horn, W., Lao, N., Murphy, K., Strohmann, T., Sun, S., and Zhang, W. (2014). Knowledge vault: A web-scale approach to probabilistic knowledge fusion. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM.

[Gardner et al., 2013] Gardner, M., Talukdar, P. P., Kisiel, B., and Mitchell, T. (2013). Improving learning and inference in a large knowledge-base using latent syntactic cues. In *EMNLP 2013*.

[Gardner et al., 2014] Gardner, M., Talukdar, P. P., Krishnamurthy, J., and Mitchell, T. (2014). Incorporating vector space similarity in random walk inference over knowledge bases. In *EMNLP 2014*.

[Hegde and Talukdar, 2015] Hegde, M. and Talukdar, P. P. (2015). An entity-centric approach for overcoming knowledge graph sparsity. In *EMNLP 2015*.

[Mitchell et al., 2015] Mitchell, T. M., Cohen, W. W., Hruschka Jr, E. R., Talukdar, P. P., Betteridge, J., Carlson, A., Mishra, B. D., Gardner, M., Kisiel, B., Krishnamurthy, J., et al. (2015). Never ending learning. In *AAAI*.

[Mnih et al., 2015] Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., et al. (2015). Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533.

[Nimishakavi et al., 2016] Nimishakavi, M., Saini, U. S., and Talukdar, P. (2016). Relation schema induction using tensor factorization with side information. *arXiv preprint arXiv:1605.04227*.

[Samadi et al., 2016] Samadi, M., Talukdar, P. P., Veloso, M. M., and Blum, M. (2016). Claimeval: Integrated and flexible framework for claim evaluation using credibility of sources. In *AAAI*.

[Sharma et al., 2017] Sharma, A., Parekh, Z., and Talukdar, P. (2017). Speeding up reinforcement learning-based information extraction training using asynchronous methods. In *EMNLP 2017*.