Crowdsourced Stock Clustering through Equity Analyst Hypergraph Partitioning

John Robert Yaros and Tomasz Imieliński
Department of Computer Science
Rutgers University
Piscataway, NJ 08854-8019
Email: {yaros,imielinski}@cs.rutgers.edu

Abstract—Use of industry classifications in the finance community is pervasive. They are critical to deriving a balanced portfolio of stocks and, more broadly, to risk management. Businesses, academics and government agencies have all researched and developed various schemes with mixed success. Recognizing major brokerages and research firms tend to assign their analysts to cover highly similar companies, we propose a scheme that makes use of stock analyst coverage assignments. Although creating coverage groups of highly similar stocks is not the direct goal of research firms, it may be imperative to their success because increasing similarity in coverage helps maximize synergy and derive the most value per analyst. To create our industry scheme, we construct a hypergraph where vertices represent stocks and hyperedges represent analyst coverage, connecting his/her similar companies. Using no additional information, we perform hypergraph partitioning to form clusters of stocks. Our scalable scheme can produce any number of clusters and can automatically update as research firms change analyst coverage as opposed to today’s leading industry schemes which have only fixed numbers of industries and require periodic expert review. Can our crowdsourced scheme match the quality of stock groups from the expert-driven schemes? We make head-to-head comparisons to a leading academic and a leading commercial creation of a single scheme over a larger universe of stocks. Amongst academics, use of Standard Industry Classification (SIC) and Fama-French (FF) groupings [2] is common. Many other schemes exist. Several data providers have devised and market industry classification schemes. One leading scheme is the Global Industry Classification Standard (GICS), devised by Standard and Poor’s (S&P) and MSCI. It has 4 levels of granularity: sector, industry group, industry and sub-industry. For example, International Business Machines (IBM) is in the “Information Technology” sector, “Software & Services” industry group, “IT Services” industry and “IT Consulting & Other Services” sub-industry. See figure 1 of section III-B. These levels form a taxonomy that allow the user to choose the appropriate level of granularity for their use. GICS is widely used amongst finance practitioners, and its general acceptance is evident by the success of the Sector Spider exchange traded funds (ETFs), whose constituent stocks are formed by GICS sectors. Investors frequently view their portfolios by percentage holdings in each industry to help them identify any under- or over-exposure to a single area of the economy. In another example, a trader may wish to hedge a long (short) position in a single stock by creating an offsetting short (long) position in a set of highly similar stocks so that if an event impacts the industry of the single stock, the trader will make up any lost money in the single stock by gains in the set of similar stocks.

The main idea of this paper is to use the coverage assignments of the major brokerages to derive groups of similar stocks. Although these research firms do not compete directly on creating stock groups, it is certainly in their best interest to have each of their analysts covering a set of highly similar stocks. We wish to combine these groupings across research firms for two reasons. First, not all research firms cover all stocks, so combining groups across research firms will allow creation of a single scheme over a larger universe of stocks. Second, and more importantly, we wish to create a consensus view with the intuition that a combination of experts should outperform single experts. That is, we “crowdsource” stock groups.

We create a hypergraph where stocks are vertices and each analyst’s coverage forms a hyperedge. We perform min-cut partitioning to derive stock groups. Intuitively, the best stock partitions will cut the fewest edges because the analyst coverage should span the fewest number of partitions. Using
this method, the user can easily define a desired level of granularity by specifying the number of groups, unlike GICS and many other schemes that offer different granularity only at a fixed number of levels. Another benefit is that our scheme can automatically update groups as analyst coverage changes, unlike GICS which requires experts to make periodic reviews. Moreover, other schemes are determined by a single organization, like S&P, whereas our scheme is crowdsourced from numerous research firms.

To evaluate the cohesiveness of these stock groups, we use a measure from Chan et. al. [3], which captures the difference between intra-group and inter-group stock price co-movement. We compare the results against GICS, FF, and a clusterer that uses past correlations to form groups. To make fair comparisons, we force our scheme to match both in number of groups and in entropy.

II. RELATED WORK

Hypergraph partitioning (HP) has been used in a variety of applications, with some of the most significant being VLSI layout, linear algebra and boolean satisfiability [4]. Strehl and Ghosh [5] present HP as an ensemble method, which is similar to our use of HP since research firms can be considered experts and their analyst coverage assignments considered their clusters. In a work that also makes use of hypergraphs, Han et. al. [6] use a frequent itemset framework, wherein they construct itemsets as all stocks that move in the same direction with magnitude greater than 2% or 1/2 point on a single day. This leads to a graph with stocks as vertices and edges formed from the itemsets. They claim to discover clusters nearly matching four sectors of the S&P 500.

Other works have examined clustering stocks without the use of HP. Doherty et. al. [7] use TreeCNG, a hierarchical topological clustering algorithm, over the close price timeseries of 73 of the FTSE 100 stocks and find their algorithm can extract clusters matching groups from the FTSE Global Classification Scheme. Each of these works, including Han et. al., use stock price co-movement to determine clusters and use expert labeled industry taxonomies to verify correctness. In contrast, our work combines groupings from research firms in order to derive clusters. We use stock price co-movement to compare our clusters head-to-head with the expert taxonomies.

Although not the work’s focus, Ramnath [8] considers the use of analyst coverage to derive stock groups. Similar to our motivation, he suggests “brokerage houses will attempt to minimize an analyst’s information acquisition costs by assigning similar firms for coverage to that analyst.” He creates groups on the basis of having at least 5 analysts following the same set of stocks. However, a large number of stocks may be unclassified because they lack the coverage to meet the threshold. Conversely, a threshold too low may lead to clusters heavily influenced by noise. Questions also arise about how to group a stock X that shares 5 analysts each with stocks Y and Z, but they are not the same analysts, so Y and Z do not have 5 analysts in common. HP more elegantly avoids these problems with the caveat that a stock still must be covered by at least one analyst.

III. PRELIMINARIES

A. Hypergraph Partitioning

A hypergraph is similar to a graph, except edges can connect any number of vertices, whereas edges in a graph each connect exactly two vertices. Formally, a hypergraph $H = (V, E)$ where $V$ is a set of vertices and $E$ is a set of edges such that for all $e \in E$, $e \subseteq V$. In other literature, edges are sometimes called hyperedges, nets or links. The vertices that compose an edge are called its pins.

$\Pi = \{V_1, V_2, \ldots, V_K\}$ is a $K$-way partition of $H$ iff [9]

- each part $V_k$ is a nonempty subset of $V$, i.e. $V_k \subseteq V$ and $V_k \neq \emptyset$ for $1 \leq k \leq K$.
- parts are pairwise disjoint, i.e. $V_k \cap V_\ell = \emptyset$ for all $1 \leq k, \ell \leq K$.
- union of $K$ parts is equal to $V$, i.e. $\bigcup_{k=1}^{K} V_k = V$.

A balance constraint is typically imposed to avoid degenerate cases where the least connected vertices are simply placed in their own part. An example balance constraint from [10] for a 2-way partition is given by a ratio $r$ and error tolerance $\epsilon$ such that $0 < r < 1$ and

$$r - \epsilon \leq \frac{|V_1|}{|V_1| + |V_2|} \leq r + \epsilon$$

In a partition $\Pi$ of $H$, an edge $e$ is said to be cut if the vertices composing the edge appear in more than one part (i.e. $\exists v_i, v_j \in e, v_i, v_j \in e \ni i \neq j, v_i \in V_k, v_j \in V_\ell, k \neq \ell$). The hypergraph partitioning problem is the task of finding a partition that minimizes the number of cut edges\(^1\) (i.e. to find $\arg \min_{\Pi} C(\Pi)$, where $C(\Pi)$ is the count of cut edges by $\Pi$).

Because the graph partitioning problem under balance constraints is known to be NP-hard [11], the same is true of hypergraph partitioning. However, various approximation algorithms have been developed. The Kerninghan-Lin (KL) algorithm [12] for 2-way graph partitioning iteratively swaps two vertices between the partitions, reducing a cost function based on the number of cut edges. The algorithm ends when it converges to a local minimum. The Fiduccia-Mattheyses (FM) algorithm [10] improves on the KL algorithm by considering single vertex moves and, for each pass, locking vertices after they have been moved. These concepts, along with efficient data structures, reduce the time for a single pass to be linear in the number of pins. FM has been further improved by the use of multi-level framework consisting of three phases: coarsening, top-level partitioning and uncoarsening. During coarsening, vertices are clustered or merged into smaller groups and edges are collapsed to connect the smaller vertex set. Depending on the coarsening algorithm, several iterations of size reductions may occur. During top-level partitioning, partitions are typically initialized randomly, then FM is used to make improvements. In the uncoarsening phase, the groups

\(^1\)Weights are commonly assigned to edges, in which case the hypergraph partitioning problem is to minimize the sum of weights of the cut edges. For this work, only unweighted edges are used.
from the coarsening phase are de-merged with vertices (or sub-groups) placed into their group’s partition from the coarsened hypergraph. The FM algorithm is used at each de-merging step as the coarsened hypergraphs of each level are unwound. This multi-level framework greatly improves runtime for large hypergraphs, while providing good solutions. See [4] for a more elaborate description of KL, FM and multi-level FM (MLFM) and a short survey of other partitioning methods.

B. Industry Taxonomies

The Standard Industrial Classification (SIC), designed by the US government, was the predominant industry classification in much of the 20th century. It is in decline amongst financial practitioners because of the lack of agreement in code assignments by data vendors [13] and because their design was more intended for government statistical use than private investing. With some of these issues in mind, Fama and French (FF) present a re-mapping of SIC codes into 48 industry groups in their study of industry costs of capital [2]. French provides additional mappings to 5, 10, 12, 17, 30, 38 and 49 groups through his data library [14]. The FF scheme is intended to form groups that are likely to share risk characteristics and often appears in academic studies [15].

Targeting financial practitioners, S&P and MSCI devised GICS, which was announced in 1999 and replaced S&P’s previous industry classification methodology in 2001 [16]. The methodology classifies companies “based primarily on revenues; however, earnings and market perception are also considered important criteria for analysis” [17]. As seen in figure 1, a single hierarchy is devised based on 8-digit code that can be used to group companies at the sector, industry group, industry and sub-industry level using the first 2, 4, 6 and 8 digits of the code, respectively. This hierarchy is applied to companies globally in order to simplify cross-border comparisons.

Bhojraj et. al. [15] compare SIC, NAICS (a government replacement for SIC), FF and GICS by computing, for each stock group in each taxonomy, an average timeseries for stock price returns and for 7 financial ratios, such as price-to-earnings or return-on-equity. Regression is preformed for each stock against its group’s average and the taxonomies are compared by their average $R^2$ values. They find higher $R^2$ for GICS in nearly every aspect of the comparison and conclude that GICS is a superior classification system.

Chan et. al. [3] provide an alternate methodology that computes the difference between the pairwise intra-group and inter-group correlation, with a higher difference attributable to better groupings (see section V-B). They suggest this approach is more applicable to portfolio analysis and risk management. Their work compares FF, GICS and a hierarchical clustering algorithm that groups stocks on the basis of a 5 year history of returns correlations. They find hierarchical clustering performs better in the training period, but underperforms when used in subsequent periods. Their results show GICS achieves the highest difference in correlation with the fewest number of groups, thus outperforming FF and historical clustering. Using a similar methodology, Vermorken [18] compares GICS with its main commercial competitor, the Industry Classification Benchmark (ICB), which is the product of Dow Jones and FTSE. Though differences exist, he ultimately finds the taxonomies largely similar for his sample of large-cap stocks.

Given these results, we focus our comparisons on GICS, FF and a correlation history based clusterer.

IV. Data

We use Compustat, a division of S&P, to select our samples of stocks, the S&P 500, 400 and 600 indices, representing large-cap, medium-cap and small-cap U.S. companies, respectively. Due to client demand, research firms are less likely to cover smaller companies, making sparsity more relevant with the S&P 600. The composition of each of the S&P indices is dynamic through time and we base our analysis on their constituents at the beginning of each time period. Unlike Chan et. al. [3], we include all constituents regardless of price levels or whether they maintain their S&P index membership, or even listing, throughout each time period. Their inclusion mitigates survivorship bias [19] and is accomplished by weighting each stock by the number of days it is listed (see section V-B).
We also obtain GICS industry codes from Compustat. Both the company assignments and the GICS taxonomy itself are reviewed and possibly updated periodically to reflect changes in the market. Hence, both the GICS groups and stock assignments are dynamic. Because we are interested in the classification schemes for the purposes of prediction (performance will be judged on future, not past, correlations), we use the historic code assigned to each company at each time step. GICS was only fully implemented in the US in 2001, but offers an extended history dating back to 1985 for S&P 500, 1992 for S&P 400 and 1995 for S&P 600 companies. The codes were assigned retroactively, so it is possible to consider them unfair for comparison. However, a high degree of similarity between GICS and its predecessor, the S&P U.S. Sector Indices, has been shown [16], so we include the extended history with the opinion that it would only raise the bar for our scheme.

The SIC mappings for the FF hierarchy are obtained from French’s Data Library [14]. The FF scheme originated in 1997 with the publication of [2], and was updated in 2004. For years 1985 to 2004, we use the original code and for years 2005 to 2010, we use the update. Like GICS, FF has the benefit of retrospect for years 1985 to 1996, but we still consider those years in our comparison. Consistent with their methodology [14], [15], we use historical SIC codes from Compustat when available, and the Center for Research in Security Prices (CRSP) otherwise.

For stock returns, we use CRSP’s comprehensive history of daily stock prices. CRSP offers pre-computed daily total return for each stock which includes not only the return from the price change from the previous days close to the current, but all other payouts to the stock holder as a result of corporate actions (e.g. cash dividends). Returns to stocks that are delisted are also used, which prevents upward biases in return calculations. Unlike Bhgoraj et. al. [15] and Chan et. al. [3] who use monthly returns, we use daily returns, which allow a more granular look at correlation differences. We can slice our history into time periods of a year, giving us more data points for statistical measures.

The actual stock coverage assignment for each analyst in each research firm is not easily obtainable. However, earnings estimates from major research firms and their analysts have been recorded by the Institutional Brokers’ Estimate System (I/B/E/S) since 1976. For each of the companies a analyst covers, s/he will approximate the quarterly and/or annual earnings the company will report in its regulatory filings. Because earnings estimates are typically made for each quarter and the analyst may make several revisions up to the company’s announcement date, estimates are a strong proxy for analyst coverage. I/B/E/S assigns unique identifiers to each analyst so that s/he may be tracked throughout his/her career, regardless of changing names or switching firms. A unique identifier is similarly assigned to each research firm. Using these identifiers, we construct our hyperedges as the set of stocks for which the analyst-firm pair provided at least one earnings estimate in the past year. Note that an analyst may change firms, but we consider this a separate hyperedge because the stocks assigned to the analyst may be different at his/her new firm. Even though it is likely the analyst will maintain a focus on the same industry at the new firm, we want our hyperedges to reflect the coverage assignments (i.e. stock universe partitioning) from each research firm separately.

The biggest drawback to our method is the potential that there are no earnings estimates for some stocks. In our sample, this occurs for one of two reasons. First, the firm was restrucuted (e.g. merger, new listing, etc.) near the end of the year and no estimates had been issued yet. Second, the company was simply not covered by any analysts. Although it may be possible to use estimates from former listings in the first case, we deemed it best to simply omit the stocks from our sample. Over our entire time period to 2010, the instances of stocks omitted were 49 / 11921 (0.41%) for the S&P 500, 50 / 7603 (0.66%) for the S&P 400 and 300 / 9598 (3.13%) for the S&P 600. As future work, we believe this shortfall might be overcome by considering a hybrid approach where analysts are used to enhance existing schemes, or existing schemes are used to place stocks without coverage.
V. Model

A. Partitioning

We partition our hypergraph using hMETIS, version 2.0pre1, a widely used Multilevel Fiduccia-Mattheyses partitioner with origins in VLSI design [20]. hMETIS offers a variety of options and we generally use defaults, with one notable exception: we enable edge reconstruction. In order to create $K$ partitions, hMETIS performs recursive bi-partitioning. By default, the cut edges are removed from the resultant two hypergraphs after each iteration of partitioning and are not considered in the next recursive call. With edge reconstruction, a new edge connecting the remaining vertices of the cut edge is added to each resultant hypergraph. We believe the link between stocks from an analyst’s coverage should be preserved even when only a subset of that coverage is being partitioned.

B. Evaluation

Following the methodology of Chan et. al. [3], we evaluate the cohesion of our clusters on the basis of similarity of stock return co-movement. We expect higher returns correlation between two stocks that each belong to the same cluster than if they had belonged to different clusters. We compute these correlations as follows. Let $I$ denote an industry (a.k.a. stock group, partition, cluster). The average pairwise correlation $\rho_{ij}$ for stock $i$ in $I$, and the average pairwise correlation $\phi_{ij}$ between stock $i$ and stocks not in its industry $I$, are

$$\rho_{ij} = \frac{\sum_{j \in I, j \neq i} d_{ij} \cdot \rho_{ij}}{\sum_{j \in I, j \neq i} d_{ij}} \quad \phi_{ij} = \frac{\sum_{j \notin I} d_{ij} \cdot \rho_{ij}}{\sum_{j \notin I} d_{ij}}$$

where $\rho_{ij}$ is the Pearson correlation coefficient between returns for stocks $i$ and $j$, and $d_{ij}$ is the number of days both $i$ and $j$ are active. To avoid survivorship bias, we include the timelinesses of delisted stocks, weighting them by the fraction of the time period they are present. This day-weighting deviates from Chan et. al. who consider only stocks that are present for the entire time period. The average intra-industry correlation $\bar{\rho}_I$ and inter-industry correlation $\bar{\phi}_I$ for industry $I$ are:

$$\bar{\rho}_I = \frac{\sum_{i \in I} \rho_{il}}{|I|} \quad \bar{\phi}_I = \frac{\sum_{i \in I} \phi_{il}}{|I|}$$

where $|I|$ is the count of stocks in industry $I$. A good classifier will maximize $\bar{\rho}_I$ and minimize $\bar{\phi}_I$.

We consider two ways to aggregate across industries. The first $\psi$ is a simple mean of the difference between intra- and inter-industry correlations. The second $\theta$ is weighted by industry size:

$$\psi = \frac{\sum_{I \in \mathcal{I}} (\bar{\rho}_I - \bar{\phi}_I)}{|\mathcal{I}|} \quad \theta = \frac{\sum_{I \in \mathcal{I}} |I| \cdot (\bar{\rho}_I - \bar{\phi}_I)}{\sum_{I \in \mathcal{I}} |I|}$$

where $\mathcal{I}$ is the set of all industries. To a user, $\psi$ conveys the average cohesion of any industry in the classification scheme, whereas $\theta$ conveys the average cohesion of any stock with its industry.

The simple mean $\psi$ is highly susceptible to imbalanced groups. Consider a disingenuous scheme that creates many tiny groups, each of highly correlated stocks, then places uncorrelated stocks into a single large group. Without change to the number of groups, $\psi$ will be higher. For this reason, $\theta$ may appear to be a superior measure since each stock has equal contribution. However, a user may be concerned with an industry-view without regard to individual stocks. Such a view is still best represented by the simple mean $\psi$. To counteract the effects of unbalancing, we compute entropy as

$$H = \sum_{I \in \mathcal{I}} \frac{1}{|I|} \log_2 |I|$$

and force our scheme to match in entropy and in number of groups when comparing with $\psi$, even though it may lead to a less natural partitioning of analysts.

hMETIS does not offer entropy constraints directly, instead providing a UBfactor, $b$. Each time a hypergraph with $n$ vertices is bisected, the two partitions will contain between $(50 - b)n/100$ and $(50 + b)n/100$ vertices. For example, if a hypergraph is bi-partitioned with $b = 5$, each partition will have between $0.45n$ and $0.55n$ vertices. In a 4-way partition, the partition sizes will be between $0.45^2n = 0.20n$ and $0.55^2n = 0.30n$ (see [21]). Because the UBfactor ultimately only describes the possible imbalance due to the smallest or largest partition, we consider it inferior to entropy, which describes imbalance across all partitions. (See [22] for a discussion of entropy’s effectiveness in measuring the balance in a clustering.) Thus, we perform multiple runs of hMETIS at small steps over a range of UBfactor values (1 to 50 in steps of 0.01) and use the run with closest entropy to the competitor scheme for the $\psi$ comparison. Entropy matching is unnecessary for $\theta$, but hMETIS still requires an imbalance factor, so we use the entropy of the analysts’ best partitioning in the previous year.

We include for comparison a correlation history based hierarchical clusterer (HIST) intended to be a straw man that provides a sense of the additional value that analysts provide rather than simply using the same set of data (stock correlations) in the past to predict the future. The clusters are created using Matlab R2010a’s `clusterdata()` with complete linkage:

$$\max\{d(a,b) : a \in A, b \in B\}$$

where $d(\cdot)$ is the distance function of one minus correlation and $a$ and $b$ are the return timelinesses of two stocks. Single or average linkage are not used as they often lead to degenerate clusters (i.e. chaining).

2hMETIS offers $k$-way direct partitioning, but suggests using recursive bi-partitioning (default) for relatively small $k$ values [21]. Our exploratory experiments confirmed worse performance with $k$-way partitioning.
We use walk-forward testing [25], [26], training on the previous year to evaluate over the next year. (See figure 3a.) For HIST, it means groups are derived using the previous year’s correlations. For the hypergraph partitioner, analyst coverage from the previous year is used to derive groups. For FF and GICS, we use the structure and labels from the last trading day of the previous year.

VI. RESULTS

Consider figure 4, where rectangles indicate GICS sub-industries and ovals indicate partitions from our analyst scheme. In the illustration, our scheme has four groups. Two of the groups have clothing retailers that arguably appeal to different generations. Group 1 with American Eagle (AEO), Aeropostale (ARO), Urban Outfitters (URBN) and J. Crew (JCG) appeals to youth, while group 3 with Ann Taylor (ANN), Chico’s (CHS) and Guess (GES) appeals to an older crowd. Group 2 unites two shoe retailers, Footlocker (FT) and Collective Brands (PSS - Payless Shoes). The partitioner places Coldwater Creek (CWTR) in group 4 with retailers not focused on clothing, but still found in malls: Fossil (FOSL), American Greetings (AM) and Regis (RGS). The GICS Apparel Retail sub-industry has average pairwise correlation of 0.484, while the groups 1, 2, 3 and 4 have correlations 0.495, 0.546, 0.598 and 0.397, respectively. Group 4 is formed across GICS sub-industries and has lower correlation, but amongst its peers in the Apparel Retail sub-industry, CWTR has the lowest average correlation (0.365), so it is reasonable to place it elsewhere. Overall, \( \Psi \) and \( \Theta \) are improved in our scheme over GICS, even when accounting for the variations caused to the stocks in the other GICS sub-industries shown in the figure. Situations such as these arise frequently and naturally with the analyst data, and make our scheme competitive.

Table I contains a summary of comparisons against each competitor. Let \( \Psi_a \) denote the series of simple means \( \psi \) for the analyst scheme (i.e. \( \Psi_a = \{ \psi_{a,1986}, \ldots, \psi_{a,2010} \} \)).
Of the competitors considered, GICS is the strongest performing scheme, which is unsurprising given previous studies [3], [15]. Against GICS, our scheme has mixed results with generally better performance for the simple mean. Against FF, analyst partitioning consistently outperforms, with weakest performance in small-cap stocks. All three competitor schemes and our scheme have higher correlation differences in the simple mean $\psi$ than the weighted mean $\theta$ (see figure 5), implying the smaller stock clusters of each scheme tend to be more highly correlated. This reinforces the need for entropy-matching for simple mean comparisons.

Our scheme tends to perform worse when fewer groups are involved, but better with more groups. This is particularly noticeable with GICS and the historical clusterer. Still, our method performs rather well at any granularity level, which supports our claim that our method is scalable, unlike the GICS and FF schemes, where the granularity of groupings is limited to a fixed set.

Since intuition might indicate that historical correlation would be the best predictor of future correlation, the reader may find most striking that the historical clusterer is not very competitive. While we acknowledge better methods may exist for grouping stocks based on historical timeseries rather than our off-the-shelf agglomerative clusterer, our results echo those found in previous work [3] where the expert taxonomies, particularly GICS, outperform groups formed by statistical clusterers.

As a side test, we consider an algorithm that generates partitions of random size and assigns stocks to each at random, but matches the number of groups and the entropy of a competitor scheme. In our trial of 1000 iterations, the partitions generated never had higher correlation difference than any of

<table>
<thead>
<tr>
<th>Scheme</th>
<th>S&amp;P 500 (Large-Cap)</th>
<th>S&amp;P 400 (Mid-Cap)</th>
<th>S&amp;P 600 (Small-Cap)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Simple</td>
<td>Weighted</td>
<td>Simple</td>
</tr>
<tr>
<td>FF-5</td>
<td>✓ 6.93</td>
<td>✓ 7.04</td>
<td>✓ 9.93</td>
</tr>
<tr>
<td>FF-10</td>
<td>✓ 4.95</td>
<td>✓ 3.70</td>
<td>✓ 7.08</td>
</tr>
<tr>
<td>FF-12</td>
<td>✓ 5.00</td>
<td>✓ 5.10</td>
<td>✓ 9.74</td>
</tr>
<tr>
<td>FF-17</td>
<td>✓ 7.71</td>
<td>✓ 7.71</td>
<td>✓ 17.8</td>
</tr>
<tr>
<td>FF-30</td>
<td>✓ 7.74</td>
<td>✓ 5.65</td>
<td>✓ 12.0</td>
</tr>
<tr>
<td>FF-38</td>
<td>✓ 8.44</td>
<td>✓ 8.41</td>
<td>✓ 8.94</td>
</tr>
<tr>
<td>FF-48</td>
<td>✓ 6.33</td>
<td>✓ 7.25</td>
<td>✓ 10.4</td>
</tr>
<tr>
<td>GICS-2</td>
<td>☒ 4.18</td>
<td>☒ 6.36</td>
<td>✓ 0.24</td>
</tr>
<tr>
<td>GICS-4</td>
<td>✓ 1.51</td>
<td>✓ 4.68</td>
<td>✓ 7.80</td>
</tr>
<tr>
<td>GICS-6</td>
<td>✓ 3.50</td>
<td>✓ 2.62</td>
<td>✓ 6.58</td>
</tr>
<tr>
<td>GICS-8</td>
<td>✓ 1.01</td>
<td>✓ 0.01</td>
<td>✓ 3.71</td>
</tr>
<tr>
<td>HIST-2</td>
<td>☒ 0.92</td>
<td>☒ 1.79</td>
<td>✓ 1.05</td>
</tr>
<tr>
<td>HIST-4</td>
<td>✓ 3.62</td>
<td>✓ 1.77</td>
<td>✓ 2.74</td>
</tr>
<tr>
<td>HIST-8</td>
<td>✓ 6.80</td>
<td>✓ 1.41</td>
<td>✓ 5.98</td>
</tr>
<tr>
<td>HIST-16</td>
<td>✓ 8.97</td>
<td>✓ 7.68</td>
<td>✓ 9.33</td>
</tr>
<tr>
<td>HIST-32</td>
<td>✓ 12.2</td>
<td>✓ 11.7</td>
<td>✓ 12.2</td>
</tr>
<tr>
<td>HIST-64</td>
<td>✓ 11.0</td>
<td>✓ 13.8</td>
<td>✓ 12.7</td>
</tr>
<tr>
<td>HIST-128</td>
<td>✓ 7.45</td>
<td>✓ 14.5</td>
<td>✓ 7.72</td>
</tr>
</tbody>
</table>

TABLE I: Summary. ✓ and ☒ indicate our analyst scheme outperformed or underperformed, respectively. *, ** and *** indicate 5%, 1% and 0.1% levels of significance in a one-tailed paired t-test. FF-X and HIST-X each refer to their respective schemes with X groups. GICS-Y uses the first Y digits of each stock’s GICS code to form groups.

Fig. 5: Example correlation difference by year. The top and bottom of each bar represent the inner-group and outer-group correlations, respectively. Longer bars indicate a higher correlation difference and, thus, better performance. For this small example from the S&P 500, analysts do better in the simple mean, but not in the weighted mean. We provide this figure to illustrate the manner of comparison being performed through the evaluation methods described in section V-B.
the schemes presented, which indicates all the schemes do contain some useful information.

Akin to the strengths of market competition over central planning, our method harnesses the competitive forces amongst research firms that drive them to efficiently assign their analysts to cover stocks. Our results exceed the expert-designed systems in several comparisons. In others, it has mixed results. Still, our method provides several advantages even if it underperforms in the correlation measure. First, our method has scalability benefits in that the user can specify any desired number of groups and their imbalance, whereas expert systems are limited to a fixed set. It is also scalable in that any number of opinion makers could potentially be added to the hypergraph. Second, our method is more automated because analyst coverage naturally adjusts to economic changes, whereas expert systems require periodic review. Third, our scheme is a means of crowdsourcing amongst many industry professionals to gather a consensus view, as opposed to other schemes which are generally produced by a single organization. We believe the method should be considered for further research and possibly use in practice.

VII. SUMMARY AND FUTURE DIRECTIONS

We present a method of constructing stock groupings that uses hypergraph partitioning to combine stock analyst coverage assignments of research firms. The output is a crowdsourced consensus view of stock groupings. Using a quality measure that compares intra- and inter-group correlation, we evaluate our method against a correlation history based clusterer and two leading expert-driven schemes, FF and GICS. In head-to-head matchups using the competitor scheme’s number of groups and entropy, our scheme generally outperforms, whereas expert systems require periodic review. It can produce any number of groups desired by the user, unlike expert schemes which offer a fixed set. It can also incorporate more opinion makers by simply adding edges to the hypergraph. Thus, our scheme is both more automated and more scalable.

As future work, we believe entropy constraints might be built directly into the partitioning algorithm rather than performing multiple runs with the imbalance factor like we do in this work. Another consideration is if companies should fractionally be assigned to multiple industries (as in the Barra USE3 model). It can be more cumbersome for a human to interpret than the single industry assignments from this work, but can be more effective in risk management, reflecting that companies frequently span multiple economic areas. For example, the 3M Company makes products from office notes to household cleaners to medical devices. It may be more appropriate to split its membership into several categories. Finally, we speculate there are other domains where the assignments of individuals to certain objects can be used to form meaningful clusters in a similar manner.

REFERENCES