MACHINE LEARNING WITH THE SUMZERO DATA FEED

The idea of "alpha capture" is not new. SumZero's community has created a large data repository that can act as a standalone, crowdsourced alpha capture system. The SumZero Data Feed can also augment and improve existing, internal alpha capture systems. We'll show you how we've built various AI model selections that aim to help investors—both quantitative and fundamental—identify outperformance derived from fundamentals-driven investment research.

We already have evidence to suggest that certain samples of individuals can have significant stock-picking skills (Gray 2008). We believe that an ensemble of machine-learning models attempting to select the best ideas from that sample can significantly increase the performance of the initial sample, on average and over time. We also know that certain ideas on SumZero are likely to produce excess returns, particularly when an idea is contrarian to sell-side recommendations (Crawford, Gray, et al. 2011). Our research has confirmed this and operationalized it. When combined with SumZero ideas, a number of other quantitative data fields like FCF Yield, Net Debt to EBITDA, and various idea text components have proven to be generalized indicators of future outperformance of an idea.

We can treat SumZero historical submissions (the Data Feed) as training data for machine learning models that learn to pick the best, new investment ideas on SumZero as soon as they are posted (as measured by its likelihood to outperform an index over an idea's predefined duration). Think of the process as a quantitative filter over qualitative rigor. This quantitative filter is not about choosing factors and weighting heavily towards "value" or "momentum." It's not about creating any rules at all. It's about providing all of the relevant (and properly normalized & neutralized) fundamental, pricing, and text data to a machine learning model and allowing the model to come up with its own investing heuristics. The result of the process is smarter and more informed portfolio management (ML models can optimize bet sizing) with greater portfolio concentration (ML model filtering reduces the number of total investments). The process looks a bit like this:

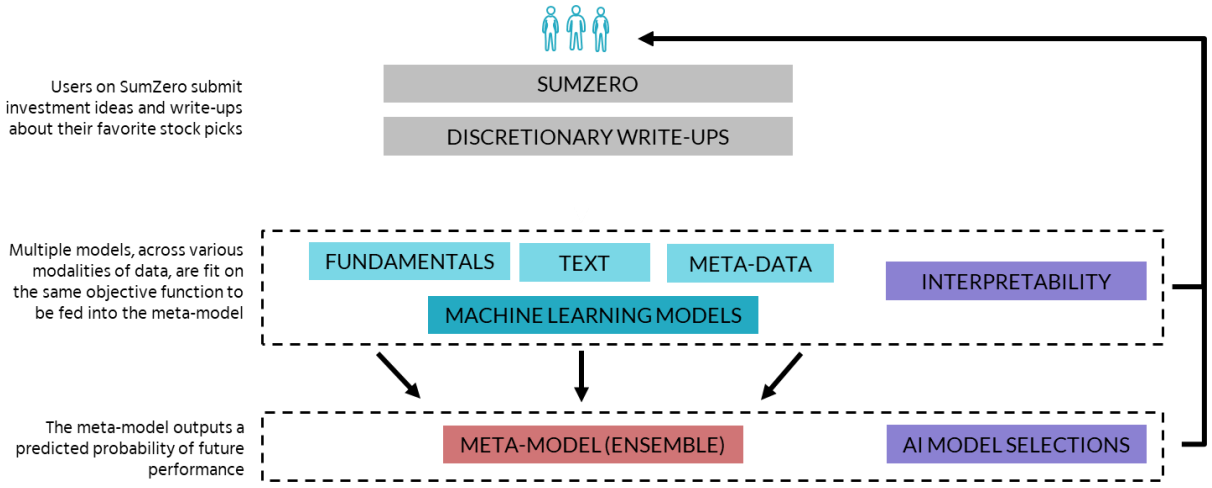

Figure 1: AI Model Selection Process

In the fundamentals data category, we download traditional valuation metrics along with their respective growth rates over different periods of time–normalized by market movements. We collect metrics like FCF Yield, Net Debt to EBITDA, ROE, ROIC, along with their close cousins, and associated growth rates, derivatives, historical premiums/discounts, and more. With this data, we build traditional quant models that aim to identify indicators of future outperformance.

Moreover, with text data we look at TF-IDF weightings and dimensionality reduction techniques to create "components" in order to group write-ups by various categories. One component category may represent the write-ups discussion of the management team and another may represent that the author heavily discussed the idea of relative valuation. We also have neural language models for standalone text classification. Recent advancements in the field of Natural Language Processing (NLP) allow the models to form a more holistic representation of an idea write-up before classifying the likelihood of outperformance. Neural models are built with neural networks (self-attention & transformers) and allow for the use of embeddings, which act as multidimensional representations of words, sentences, or documents. More recent neural language models are being pre-trained on enormous datasets to allow better understanding of documents than ever before.

Simplistically, the model "reads" the write-up text (using the NLP models noted above) and analyzes the quantitative data (e.g. fundamental data noted above) on the back-end to build a holistic "understanding" of the idea to then make a better informed investment decision.

For meta-data, we plan to continue to experiment with fields including user rankings, ratings, time of posting, whether an idea is a repeat of a recently posted idea, etc. We don't include any meta-data in any of our current models, but we plan to incorporate these in the future and expect improved results.

On the technological side, for general classification training we tend to use tree-based models (i.e. gradient boosted models like XGBoost, LGBM, etc) and for neural language models we use common deep learning frameworks and pretrained models (i.e. pytorch, tensorflow, huggingface). We build multiple models over different subsets of the features and ensemble the outputs of all the models together. Our entire stack is simply Python + SQL.

FRAMING THE DATA AND ML PROBLEM (BUILDING AN EXAMPLE MODEL)

The data itself spans over 12 years and consists of over 14,000 global equity ideas that were pitched to the SumZero community along with associated natural language text write-ups. Text write-ups are descriptions of why someone recommends buying (or selling) the pitched stock at that particular time. We also collect financial data and calculate trading returns for all of those write-ups over various time periods. The dataset spans from 2008 through today and is carefully corrected for issues like survivorship bias and potentially missing or incorrect data. Other pre-processing like normalization, market neutralization, and feature engineering that we perform all have a large impact on model performance.

For this example model, we'll be using the entirety of SumZero's Data Feed and attempting to select the ideas that are most likely to outperform the S&P 500 over a one-year period. First, we find a stock's 1-year return vs. the S&P 500 (stock's 1Y return – S&P500 1Y return) starting from the date that a stock was pitched to the SumZero community. If the relative return is positive, the target is labeled "1" for outperformance and if the number is negative, the target is labeled "0" for underperformance.

After downloading and cleaning the data, a sample dataframe which is just about ready to be passed through a machine learning pipeline for binary classification may look like this:

WRITE-UP INFO			FEATURES									TARGET	
Company	Ticker	Submission Date	Full Investment Write-up Text	Country of Risk	...	Debt to Equity	EV/Sales LTM	FCF Yield LTM	Net Debt to FCF	ROIC	Rating	1Y Return vs. SPX	Target (Outperformed)
Ingram Micr	IM	3/25/2008 15:23	IM is the largest global tecl	US	...	15.27	0.08	7.83	-0.20	8.46	4.75	17%	1
Pyxus Intern	PYXSQ	3/26/2008 10:17	The industry is essentially a	US	...	351.61	0.59	39.14	3.28	3.20	5.00	8%	1
Securitas AB	SCTBF	3/27/2008 10:28	Securitas AB [SECUB SS] is	SE	...	181.16	0.64	6.79	3.43	4.26	3.18	13%	1
...	...	...	...	...	...	...	...	...	...	...	...	...	...
Klabin SA	KLBN3	9/8/2011 18:15	Klabin S.A. is the biggest pr	BR	...	89.15	1.89	8.66	2.48	6.53	3.00	18%	1
Oil & Natura	ONGC	9/7/2011 18:07	ONGC is a state-owned Ind	IN	...	5.46	1.81	10.30	-0.32	15.97	4.71	-15%	0
Salem Medi	SALM	9/8/2011 22:29	LM reported AH today. Lon	US	...	143.64	1.61	27.79	10.02	3.99	5.00	98%	1
Actua Corp	ACTA	9/13/2011 15:00	Internet Capital Group, Inc.	US	...	7.17	1.71	-0.02	-27.83	-7.36	5.00	-17%	0
Regis Corp	RGS	9/8/2011 19:32	Thesis: Company is trading	US	...	30.35	0.44	19.43	0.95	0.40	3.44	13%	1
Investors Tit	ITIC	9/9/2011 15:26	Investors Title Company pr	US	...	0.00	0.72	9.30	-1.60	4.54	3.00	63%	1
Signet Jewe	SIG	9/9/2011 18:46	Signet is the dominant and	US	...	0.62	0.74	6.23	-1.63	11.26	4.57	9%	1

Figure 2: Sample Raw Data Structure

Best practice indicates the data should be split into three parts: a training set, a test set, and a validation set with no overlap in time between the sets. Using various combinations of the feature sets, we train multiple models to predict the same target: one year outperformance. Different models learn different representations of the problem and when averaged together tend to correct for errors amongst themselves.

Following successful training, we define rules for when an idea is added as a model selection. For simplicity and for this example model, if an idea scores above 50% probability of outperformance, that idea is added as a model selection. The idea is sold after one year or when the author closes the idea, whichever comes first. Using this methodology, **from January 2020 through February 2021, the ideas added to SumZero's 'Best Ideas' model has outperformed the S&P 500 by 13.17% on average (or a median of 7%).**

INTERPRETABILITY

After a model is trained and validated, we can show how the model came to a particular conclusion—opening up the "black box." We need to further ensure the model isn't overfit and its decisions make sense. From a technical standpoint, we use Shapley/SHAP values, which can be thought of as feature importance scores for interpretability of the model. SHAP values allow us to also

compute explanations for each prediction a model makes. For our purposes, **think of SHAP values as how much the model "likes" or "dislikes" a given feature.** If we combine all of the SHAP values for each investment submission, we can see a global interpretation of the model's decision-making process:

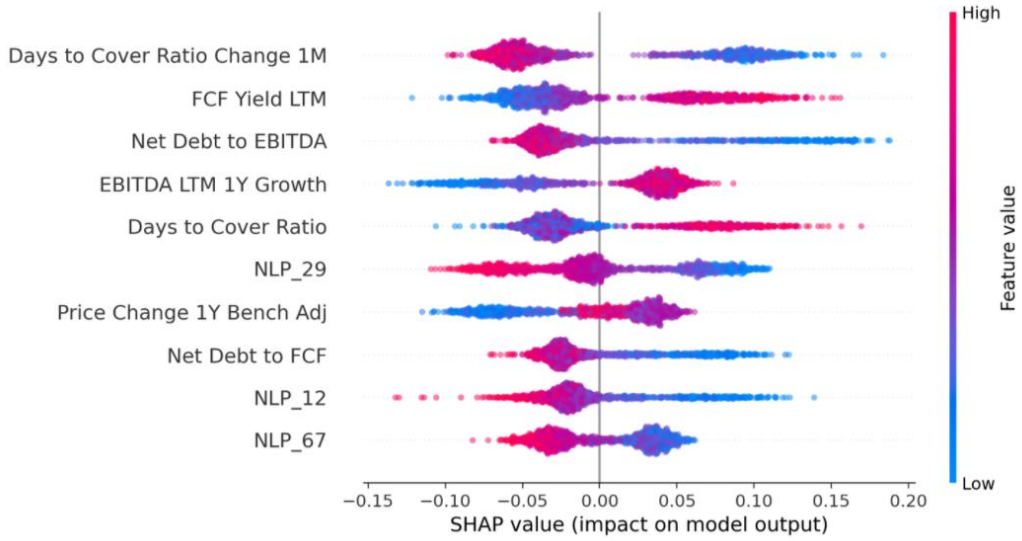

Figure 3: Global SHAP Values

The summary plot above is computed by looking at every single SHAP value for all investment write-ups over each feature. Each dot is a data point and each feature's explanation is another row on the summary plot. The different features are ordered on the y-axis (vertical) according to their total importance. The x-axis (horizontal) is the SHAP value; to the right of zero means the feature value for that datapoint had a positive impact on the model's output (more likely to outperform the market) and to the left of zero means the feature value for that datapoint had a negative impact on the model's final probability output (less likely to outperform the market). The color from blue to red represents the normalized value of the feature (factor) from low to high. A very bright red value would represent the highest value of a certain feature and a very bright blue value would represent the lowest value of a certain feature.

In the case of SumZero ideas, the model "likes" ideas that have decreases in days to cover ratio (Days to Cover Ratio Change 1M), high FCF Yield (FCF Yield LTM), low net debt to EBITDA, and high momentum relative to the market over a one year period prior to posting (Price Change 1Y Bench Adj). But, if you can learn these heuristics in any introductory investment course, why are these model decisions so novel? For one, the model figured out all of these investing heuristics on its own solely from viewing past examples. We did not teach the model these rules, it learned these lessons and many more from the data itself. These rules of thumb aren't clouded by the emotional biases that humans carry. And the model didn't just figure out the general rules like "buy high FCF yield," it has also learned multidimensional heuristics based on the interaction effects of various factors. Those relationships can also be seen by each field. Notice how those relationships are often non-linear:

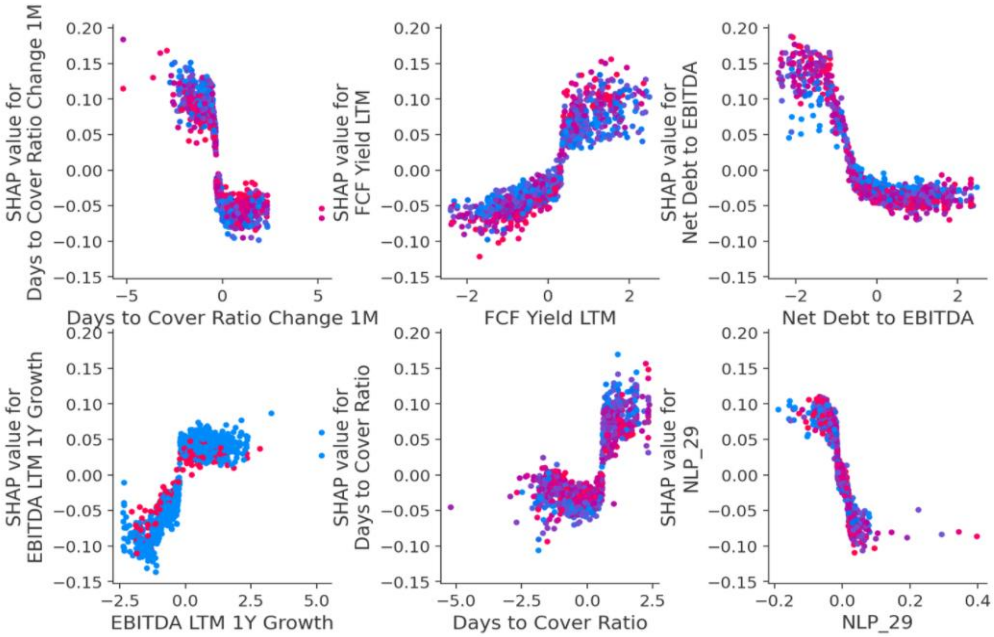

Figure 4: Global SHAP Decision Plots

x-axis is normalized

As previous research demonstrates, ideas on SumZero are likely to produce excess returns particularly when an idea is contrarian to sell-side recommendations (Crawford, Gray, et al. 2011). We've derived a feature that measures the sell-side's price target relative to the price of a stock when it is added to SumZero. Our model discovers a similar contrarian relationship as compared to these prior studies. **When the average sell-side analyst has a high price target relative to the stock's current price, our model believes that idea has a lower likelihood of outperforming the S&P 500.**

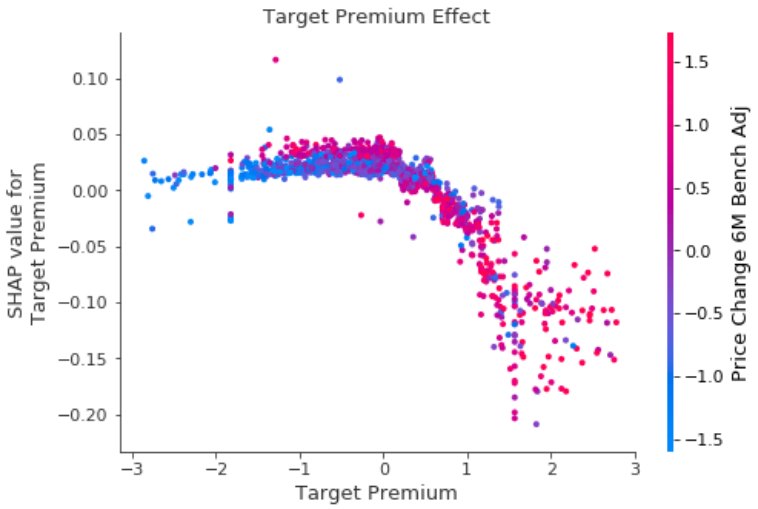

Figure 5: SHAP Decision Plot for Price Target Premium

x-axis is normalized

We also interpret what text features in a write-up are important to the model. We call a group of terms an "NLP Component." Importantly, these terms contained within a component are not ones that we have defined or chosen. These terms like "cash flow" and "balance sheet" were chosen and grouped in an end-to-end manner by our algorithms. Then, a classification model decides whether a write-up with a high number of these terms is more or less likely to outperform the market. Below is an illustration of an example component and the top terms (n-grams) that are contained within that component.

	Features	Score
0	cash flow	0.003387
1	long term	0.002540
2	balance sheet	0.002407
3	market cap	0.002210
4	buy stock	0.002158
5	total assets	0.002041
6	free cash	0.002018
7	free cash flow	0.002015
8	net cash	0.001933
9	longer term	0.001882

Figure 6: Example NLP Component

SCALABILITY AND BUILDING MORE MODELS

The best application for these models is to use them in a portfolio management pipeline, where the model systematically selects the ideas to be held in a portfolio. **There are many different portfolio options that can be built with the same underlying data (e.g. best ideas, long/short, tech ideas, SMID-cap focused, etc.). The systematic, pipeline approach is highly scalable.** In addition to training an AI model for "Best Ideas," we can (and do) train alternative models that look to optimize over different time periods, benchmarks, sectors, etc. Remember, with the data we've gathered for this example model, our target was to predict whether or not a stock outperformed the S&P 500 over a 1-year time frame. This target can be changed to whatever time horizon or benchmark an investor is interested in targeting.

These AI Model Selections and their constituents are updated daily and are provided as an add-on to the SumZero Data Feed. Alternatively, **with the underlying SumZero Data Feed and the information provided in this report, you should have all the tools at your disposal to reproduce and improve upon our internal models driving the AI Model Selections.**

NOTES AND CITATIONS

- Crawford, Gray, et al. (2011) The Investment Value of Contrarian Buy-Side Recommendations https://papers.ssrn.com/sol3/papers.cfm?abstract_id=1971533

- Gray, W. (2008) Do Hedge Fund Managers Have Stock-Picking Skills? https://papers.ssrn.com/sol3/papers.cfm?abstract_id=1477586

- Hudson & Thames (2019): https://hudsonthames.org/meta-labeling-a-toy-example/

- López de Prado, M. (2018a): Advances in Financial Machine Learning. 1st ed.

- López de Prado, M. (2018b): "The 10 Reasons Most Machine Learning Funds Fail." The Journal of Portfolio Management, Vol. 44, No. 6, pp. 120–33.

DISCLAIMERS