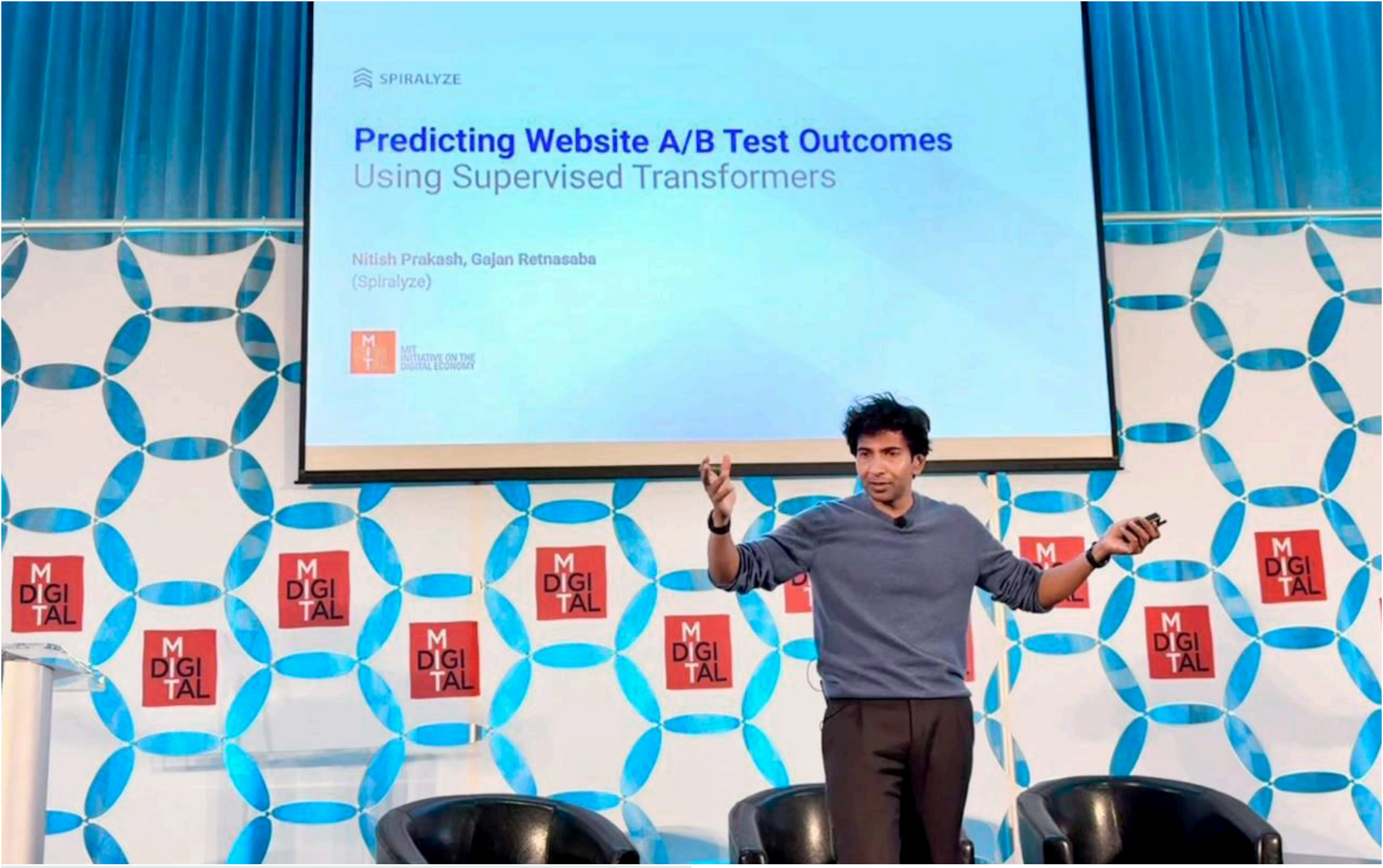


CODE @MIT

Predicting Website A/B Test Outcomes Using Supervised Transformers

Nitish Prakash, Gajan Retnasaba
nitish@spiralyze.com
Spiralyze (Atlanta, GA)

Abstract

A/B testing is the industry standard tool for improving website conversion rates, but selection of candidate tests remains largely heuristic with industry win rates rarely exceeding 30%. We present the first large-scale multimodal AI approach for **ex-ante** prediction of website A/B testing outcomes. Our novel dataset spans 73,000 historical experiments from 7,789 websites, an order of magnitude greater than prior training datasets. The model combines Vision and Text Transformers, using control-variant image pairs and OCR-extracted text content, with categorical embeddings for metadata (industry, page type, business model, customer type). The model employs a gated fusion mechanism that dynamically weights visual and textual features based on contextual metadata, enhanced with directional augmentation and quality-based sample weighting. Following contrastive pre-training and supervised fine-tuning, the model attains 62% accuracy with F1=0.62 and ROC AUC=0.66 on a held-out set of 11,038 tests. On a benchmark of 100 curated tests, the model achieved 65% accuracy with F1=0.66, outperforming (i) experienced expert practitioners (46% accuracy, F1=0.46) and (ii) multimodal LLMs (39% accuracy, F1=0.37). These findings and early pilot results show the potential for machine learning combined with large-scale experiment data to provide decision-support tools for experiment selection, and ultimately to increase experiment win rates.

Keywords: *A/B testing, self-supervised learning, supervised fine-tuning, conversion optimization*

1. Introduction

A/B testing is the primary tool used to improve the conversion rates of websites, apps, advertising, and digital experiences. In an A/B test, a control (original version) is compared against a variant (modified version) to measure which performs better on a key metric such as the visitor-to-purchaser conversion rate. While the testing methodology is rigorous, deciding which ideas to test remains a highly subjective process. Test selection is primarily based on the subjective judgment of experts and simple heuristics [1] which leads to industry A/B tests win rates between 10% to 30% [2,3]. Research on human experts find that they perform little better than chance in evaluating experiment results [4].

Research has explored the use of LLMs to predict the results of A/B tests on news headlines [5]. LLMs have also been used to simulate users and forecast A/B test results on Amazon.com [6]. Pattern libraries aggregate test results from multiple websites, cluster them into repeating "patterns" and use this pooled data to predict the performance of future tests, reportedly achieving over 80% accuracy [7]. Additionally, there are unpublished reports of AI models trained on historic data predicting the results of A/B tests on Facebook and Instagram Ads [8, 9] and email subject lines [10].

This paper introduces the first known application of a supervised multimodal AI to predict the outcomes of website A/B tests using a large dataset. The model is trained on a dataset of 62,548 and tested on 11,038 website A/B tests, the largest such dataset known to the authors by an order of magnitude [11]. By learning from the outcomes of thousands of prior tests we aim to capture patterns that elude human experts, whose predictions are constrained by limited data, recall, and inference. We benchmark the model's performance against human experts and commercially available LLMs.

2. Methodology

3.1 Dataset and Preprocessing

We compiled a dataset containing 73,688 A/B test results from 7,773 companies. To our knowledge this is the largest collection of A/B test results assembled, giving the model a huge advantage over human experts and LLMs, who only have access to the small corpus of published test data [11]. The data includes:

- **Third-party data (91%)**: 67,159 test results were scraped from public websites using a proprietary system that captures live A/B tests and imputes the

Predicting Website A/B Test Outcomes Using Supervised Transformers

Nitish Prakash, Gajan Retnasaba
nitish@spiralyze.com
Spiralyze (Atlanta, GA)

Abstract

A/B testing is the industry standard tool for improving website conversion rates, but selection of candidate tests remains largely heuristic with industry win rates rarely exceeding 30%. We present the first large-scale multimodal AI approach for *ex-ante* prediction of website A/B testing outcomes. Our novel dataset spans 73,000 historical experiments from 7,789 websites, an order of magnitude greater than prior training datasets. The model combines Vision and Text Transformers, using control-variant image pairs and OCR-extracted text content, with categorical embeddings for metadata (industry, page type, business model, customer type). The model employs a gated fusion mechanism that dynamically weights visual and textual features based on contextual metadata, enhanced with directional augmentation and quality-based sample weighting. Following contrastive pre-training and supervised fine-tuning, the model attains 62% accuracy with F1=0.62 and ROC AUC=0.66 on a hold-out set of 11,038 tests. On a benchmark of 100 curated tests, the model achieved 65% accuracy with F1=0.66, outperforming (i) experienced expert practitioners (46% accuracy, F1=0.46) and (ii) multimodal LLMs (39% accuracy, F1=0.37). These findings and early pilot results show the potential for machine learning combined with large-scale experiment data to provide decision-support tools for experiment selection, and ultimately to increase experiment win rates.

Keywords: A/B testing, self-supervised learning, supervised fine-tuning, conversion optimization

1. Introduction

A/B testing is the primary tool used to improve the conversion rates of websites, apps, advertising, and digital experiences. In an A/B test, a control (original version) is compared against a variant (modified version) to measure which performs better on a key metric such as the visitor-to-purchaser conversion rate. While the testing methodology is rigorous, deciding which ideas to test remains a highly subjective process. Test selection is primarily based on the subjective judgment of experts and simple heuristics [1] which leads to industry A/B tests win rates between 10% to 30% [2,3]. Research on human experts find that they perform little better than chance in evaluating experiment results [4].

Research has explored the use of LLMs to predict the results of A/B tests on news headlines [5]. LLMs have also been used to simulate users and forecast A/B test results on Amazon.com [6]. Pattern libraries aggregate test results from multiple websites, cluster them into repeating “patterns” and use this pooled data to predict the performance of future tests, reportedly achieving over 50% accuracy [7]. Additionally, there are unpublished reports of AI models trained on historic data predicting the results of A/B tests on Facebook and Instagram Ads [8, 9] and email subject lines [10].

This paper introduces the first known application of a supervised multimodal AI to predict the outcomes of website A/B tests using a large dataset. The model is trained on a dataset of 62,548 and tested on 11,038 website A/B tests, the largest such dataset known to the authors by an order of magnitude [11]. By learning from the outcomes of thousands of prior tests we aim to capture patterns that elude human experts, whose predictions are constrained by limited data, recall, and inference. We benchmark the model’s performance against human experts and commercially available LLMs.

2. Methodology

3.1 Dataset and Preprocessing

We compiled a dataset containing 73,688 A/B test results from 7,773 companies. To our knowledge this is the largest collection of A/B test results assembled, giving the model a huge advantage over human experts and LLMs, who only have access to the small corpus of published test data [11]. The data includes:

- **Third-party data (91%)**: 67,159 test results were scraped from public websites using a proprietary system that captures live A/B tests and imputes the

outcome based on the arm of the test deployed on conclusion of the test. This data is noisy as individual websites apply varying levels of rigor in testing and result interpretation.

- **First-party experiments (9%)**: 6,529 test results came from A/B tests run by our company across 195 client websites. This data is generally more trustworthy because we have complete visibility into not only control and variation design and outcome, but also the experiment setup, sample sizes, magnitude of result, and more.

Each A/B test record in the dataset includes -

Page Image : Screenshots of both the control and variant as presented to users.

Page Text Content: We extracted the textual content of both the control and variant pages using Optical Character Recognition (**OCR**) techniques.

Metadata: For each test, we have structured metadata describing the context:
1. Industry (e.g. finance, e-commerce, HR)
2. Page type (e.g. homepage, product page, signup page, landing page)
3. Conversion goal/type (e.g. lead gen, direct purchase)
4. Customer type (B2B, B2C)
5. Business model (e.g. e-commerce, SAAS)

These five categorical features provide context that can influence what design elements matter, for instance, a B2B SaaS landing page might rely more on informative text, whereas a B2C e-commerce product page might be more visual.

Outcome Label: Finally, each record has a binary outcome of the test: **Winner** (the variant outperformed the control with statistical significance) or **Loser** (the variant did not beat the control, which includes neutral/no-significance cases). For third-party scraped tests, we infer "winner" by tracking which version is retained in the month following the conclusion of the test. For our first-party tests, the label comes directly from the statistical analysis of the experiment.

After collection, we **split** the dataset into 90% (62,548) for training and 10% (11,038 tests) held out for evaluation. We performed several **preprocessing** steps to prepare the data for model training:

Sample Weighting: Because our data comes from mixed sources of varying reliability, we implemented a sample weighting scheme during training. In general, we assigned **higher weight** to examples that were likely more trustworthy or informative: e.g. first-party tests (which have verified outcomes) were weighted more than third-party test; tests from websites with very high traffic (where results are more likely to be significant) were weighted more than tests from low-traffic sites. These weights were used in the loss function so that the model pays more attention to high-quality data during training.

3.2 Model Architecture

We designed a Siamese multimodal neural network that ingests the pair of pages (control and variant) along with metadata and outputs a prediction of which version won. Both the control and variant go through parallel vision and text encoders, then the features are combined, and a classification head predicts win or loss. The notable components of the architecture are:

Vision and Text Encoders: For the visual content, we use a Vision Transformer (ViT-B/16, pretrained on ImageNet21k) to encode each page screenshot into a 768-dimensional visual embedding. For the textual content, we use a DistilBERT encoder (uncased, base model) to produce a 768-dimensional textual embedding for each page.

Metadata Embeddings: Each of the five categorical metadata fields (Industry, Page Type, Conversion Goal, Customer Type, Business Model) is mapped to an embedding vector.

Difference Features and Gated Fusion: We take the difference between the control and variant embeddings (and also the reverse difference variant–control, to preserve directionality). This yields a visual difference feature and a text difference feature that capture how the variant's content deviates from the control's content. Intuitively, these highlight the changes introduced by the test. We then apply a **gated fusion mechanism** that decides how much emphasis to give to the visual differences vs textual differences on a *per-test* basis.

Swap-Based Data Augmentation: We incorporate a novel augmentation during training called **swap augmentation** (or directional augmentation). In an A/B test, the roles of "control" and "variant" are not interchangeable, if we swap the two page inputs, the outcome label inverts (if variant won originally, swapping them would make it as if the opposite outcome happened). To teach the model this concept of **asymmetry**, we randomly swap the control and variant inputs for roughly 10% of training examples *and* flip the win/loss label.

Fusion and Output Layer: After encoding both pages and applying the gated modality weighting, the model concatenates all pertinent features into one long vector.

Specifically, we include: the raw visual embedding of control and variant, raw text embedding of control and variant, the **weighted** visual and textual difference embedding, the combined metadata embedding vector, and a small **role indicator** (we append a constant vector [1,0] to indicate which input was originally the control - this is another way to break symmetry, ensuring the model knows the first image/text corresponds to control). This concatenated vector goes into a final **prediction head** consisting of a multi-layer perceptron (MLP). The MLP has a few dense layers with GELU activations and layer normalization, gradually reducing to a single logit output. We apply a sigmoid to produce a probability (p) that the variant is the winner. If ($p > 0.5$), the model predicts the variant won; if ($p < 0.5$), the model predicts the control won (i.e. variant lost). We use dropouts in this head for regularization. In essence, the prediction head learns a nonlinear function of the vision features, text features, and their interactions to decide if the changes in the variant are likely to improve conversions or not.

2.3 Training Strategy

We devised a two-stage training strategy to maximise learning from the data :

Stage 1: Self-Supervised Contrastive Pre-Training
In the first stage, we trained the Siamese network **without using the win/loss labels**, instead using a contrastive learning objective to prime the model. We treated each control-variant pair as a positive pair in a contrastive setup: the model should learn to output embeddings for control and variant that are **similar to each other** (since they are two versions of the same page) but **dissimilar from other pages' embeddings**. Concretely, we employed an NT-Xent (normalized temperature cross-entropy) loss as used in SimCLR: it tries to minimize the distance between embeddings of the control and its variant while maximizing distance to all other control/variant embeddings in the batch. This pre-training forces the model to pay attention to the differences between a page and its variant, effectively learning a representation of the *change* itself without yet worrying about which direction (win or lose) the change goes. The model sees many examples of "here is an A and a B of the same experiment" and learns to encode that concept. This stage leverages all training pairs, even if their outcome labels might be noisy, since no labels are required. We trained contrastively for 50 epochs, the average loss decreased from 0.1638 in 1st epoch to 0.0099 in 50th epoch.

Stage 2 : Supervised Fine-Tuning
In the second stage, we fine-tuned the model on the actual **win vs. loss classification task**. We added the prediction head and trained with binary cross-entropy on the labeled outcome of each test. To handle the **class imbalance** (there are more losing tests than winning tests), we used a **weighted focal loss** function. The focal loss extends binary cross-entropy with a factor that down-weights easy examples and focuses on hard, misclassified examples, this is useful when majority samples are of one class (losers) and we want the model to still learn to detect the minority class (winners). We set the focal loss parameters (alpha=0.25) to give extra weight to the positive class, assuming "win" is positive and (gamma=1.5) modulating factor controlling how much to focus on difficult examples. We also incorporated the **sample weights** so that high-quality first-party tests contribute more to the loss. Another technique we employed was **label smoothing** of 5% on the targets, to prevent the model from becoming over-confident. This means if a sample is labeled 1 (win), we actually train with a target of 0.95, and if labeled 0 (loss), target 0.05 - this small smoothing can improve generalization in classification tasks with noisy labels. We fine-tuned on a subset of ~51k labeled examples (and validated on another ~11k) for up to 40 epochs (Figure 1), using early stopping on validation AUC. The transformers were fine-tuned at a lower learning rate (on the order of 2e-5) while the new layers (gating and MLP head) used a higher learning rate (around 2e-4) to learn faster. We used the AdamW optimizer with a one-cycle learning rate schedule. During fine-tuning, we continued to apply the **swap augmentation** (randomly flipping control/variant and label) on-the-fly to reinforce the directional understanding.

Through this two-stage process, the model first learned a general notion of "what makes two page versions different" and then learned "which differences tend to correlate with winning or losing." We found that doing contrastive pre-training improved the subsequent supervised performance, as opposed to training from scratch on the labels. The focal loss and weighting were important to achieve good recall on the winning class.

Figure 1. Supervised Training Graph

3. Results

3.1 Model Performance

Model performance was evaluated on the 11,038 hold out tests that the model was not exposed to. The model demonstrated 61.84% accuracy, correctly identifying the outcome of 6,826/11,038 tests. ROC-AUC is 65.77%. Notably, the model is better at predicting losing tests than it is as predicting winners.

Table1: Model Performance on Hold-Out Set of 11,038 tests

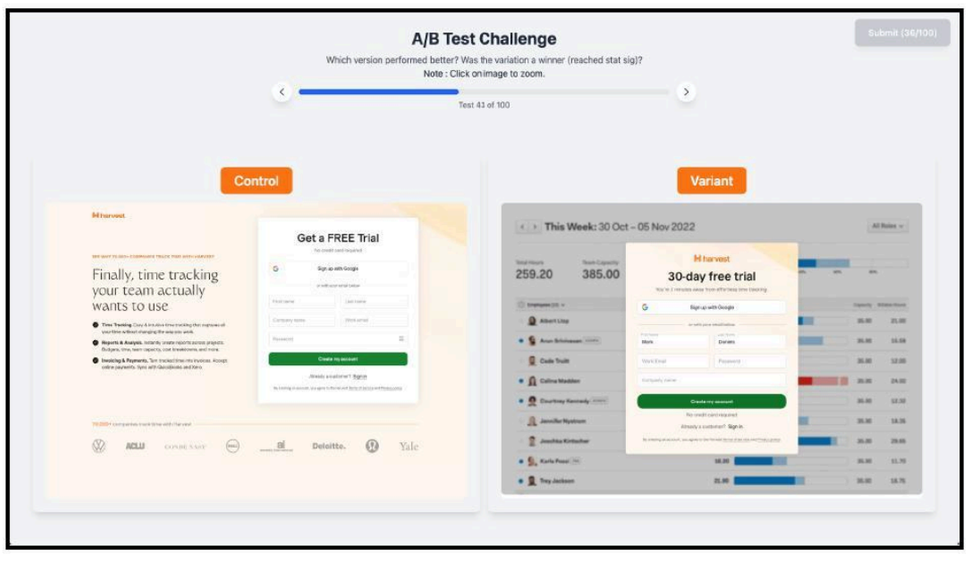
	Control	Variation	Weighted
Precision	0.68	0.55	0.62
Recall	0.65	0.58	0.62
F1 Score	0.66	0.56	0.62

3.2 Human Expert & Multimodal LLM Benchmarking

The model was benchmarked against a panel of 26 conversion rate optimization (CRO) experts. We hired four external experts via the Upwork marketplace, selecting CRO professionals with high ratings (>4.8/5), over 10 completed jobs, and rates above $100/hr. We also recruited 22 experienced CRO practitioners from our company and incentivized them with payment based on performance.

Each expert was asked to make predictions on the outcomes of 100 A/B tests. Using an online assessment, for each prediction they were shown images of the control and variation and asked to predict whether the variation won.

Figure 2. Experts were asked to predict the outcomes of 100 A/B Tests via an online assessment.



The 100 A/B tests were first-party tests drawn from our hold-out set. The experts averaged 46/100 correct predictions, with individual scores ranging from 36/100 to 63/100. The average performance of the external (45/100) and internal (47/100) experts was similar.

The model was also benchmarked against commercially available large language models (LLMs). Using their respective APIs, we prompted each LLM to predict the outcome of the same 100 A/B tests as if they were a CRO expert. Among the models tested including state-of-the-art OpenAI and Gemini models, Gemini 2.5 Pro achieved the highest score of 39/100.

The model correctly predicted 65/100 tests and also achieved the highest scores for precision, recall and F1. The outperformance was largely driven by the model's superior performance in identifying losing tests.

Table 2: Comparison of Model to CRO Experts and LLMs on a set of 100 A/B tests

Metric	**Supervised Model**	**Human CRO Experts**	**Gemini 2.5 Pro**	**Open AI 4o**
Accuracy	0.65	0.46	0.39	0.32
Weighted Precision	0.70	0.65	0.60	0.65
Weighted Recall	0.65	0.46	0.39	0.32
Weighted F1 Score	**0.66**	**0.46**	**0.37**	**0.20**

4. Discussion

This work demonstrates the potential for supervised transformers combined with large datasets to predict the outcome of website A/B tests at higher levels of performance than the current heuristic-based approach. These models have the potential to unlock major testing gains for organizations running experimentation programs.

The most immediate practical short-term application is providing AI decision support for experimentation. In a typical optimization program, teams have substantially more ideas than they have the capacity to test, and models could help them rank these ideas based on their likelihood to win and better prioritize their resources. In particular the model seems to be particularly talented at sniffing out losing tests.

We are working on aiding the decision making by giving access to users to not just model prediction but also the 10 closest historical test 'comparables'. It is hoped this will increase trust by providing the user with more context for the model's decision so that they can better decide when to

override the model's prediction.

The model also found an unexpected use in advising clients on proposed changes on pages with too little traffic to support testing. Model predictions gave clients confidence that changes could be implemented with low risk of being negative.

The findings also hint that it may be possible using the collective experience contained in millions of web experiments to create a predictive system that would foresee the likely outcome of tests and substitute for testing. First, model performance could be further improved with data from more tests and more fine-grained performance data than was available from the third-party data that comprise the majority of our dataset. This could potentially be achieved through a cooperative testing network with sites sharing anonymized data in return for access to a more powerful predictive model and thereby spreading learning costs across the network. Second, using more inputs, beyond just screenshots, could further enhance the model, capturing non-visual factors such as user demographics. Third, the approach could be extended to other domains of digital experimentation, including mobile apps, email & text campaigns.

A longer-term application of these prediction models is as a building block toward an AI that can generate high potential experiment ideas and designs.

References

[1] R. Kohavi, A. Deng, R. Longbotham, and Y. Xu. 2014. Seven rules of thumb for web site experimenters. Proceedings *of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '14)*. ACM, New York, NY, USA, 1857–1866. https://doi.org/10.1145/2623330.2623341

[2] Optimizely. 2021. Get more wins: Experimentation metrics for program success. *Optimizely Blog*. Retrieved from https://www.optimizely.com

[3] G. Georgiev. 2022. What Can Be Learned from 1001 A/B Tests? *Analytics Toolkit Blog*. Retrieved from https://blog.analytics-toolkit.com

[4] J. Linowski. 2022. Beyond Opinions About Opinions: What 70,149 Guesses Tell Us About Predicting A/B Tests. *GoodUI Blog*. Retrieved from https://goodui.org

[5] W. Kurt. 2023. Replacing an A/B Test with GPT. *Count Bayesie Blog*. Retrieved from https://countbayesie.com

[6] D. Wang, et al. 2025. AgentA/B: Automated and scalable web A/B testing with interactive LLM agents. *arXiv preprint* arXiv:2504.09723.

[7] J. Linowski. 2018. Predicting Winning A/B Tests Using Repeatable Patterns. *CXL Blog*. Retrieved from https://cxl.com

[8] M. Juetten. 2021. AI for advertising: Pattern89. *Forbes Online*. Retrieved from https://www.forbes.com

[9] SporeLogic. 2025. Horizon: The world's first predictive engine for advertising. Retrieved from https://www.sporelogic.com

[10] Braze. 2024. Q2 2024 Earnings Call Transcript. Retrieved from https://www.braze.com

[11] GoodUI. 2025. 604 A/B tests. Retrieved from https://goodui.org