Paper Review of Witan: Unsupervised Labelling Function Generation for Assisted Data Programming

Rémi Kalbe[1] and Irina Klein[2] and Ryan Manthy[3]

[1],[2],[3]Illinois Institute Of Technology

[1]rkalbe@hawk.iit.edu, [2]iklein@hawk.iit.edu, [3]rmanthy@hawk.iit.edu

Abstract -

In this review, we delve into the capabilities and performance of the Witan algorithm as presented in the original research paper. The Witan algorithm tackles the significant task of automatically generating *labeling functions* (LFs)—rules that help categorize data—without needing any initial supervised data or predefined categories.

We distill the essence of the Witan algorithm, noting especially its use of a utility function. This function, inspired by algorithms that learn from rules, cleverly selects and hones LFs using information inherent in the data itself. This process is done without relying on any pre-labeled examples, which is a remarkable step forward.

Our evaluation focuses on Witan's interactive design that allows users to effortlessly review and refine the suggested LFs. This functionality notably lessens the usual labor-intensive process of data categorization. The empirical studies recounted in the source paper illustrate Witan's adeptness at classifying data in binary and multi-class settings—and that too by asking very little from the user. The algorithm's performance closely approximates that of methods that have access to fully labeled data sets.

To wrap up, we contemplate the implications of integrating Witan into machine learning pipelines. Its ability to ease the burdens of data programming makes Witan a valuable breakthrough in the field of assisted data programming, worthy of attention for future applications.

1 Introduction

In the research paper, we're introduced to the WITAN algorithm, an innovative tool for automatically labelling large collections of data without human guidance. This clever algorithm looks at the characteristics of unlabelled data, figures out rules to categorize it, and combines these rules in different ways. A unique feature of WITAN is that it actively involves users, enabling them to explore and refine labeling rules—perfect for situations where the types of categories in the data are not known beforehand.

2 Context & Motivation

Machine learning, particularly through deep learning techniques, has seen tremendous growth thanks to the availability of vast, accurately sorted datasets for training. This is evident in achievements like recognizing what's in a photo or understanding the sentiment in a text. However, labeling data is often slow and costly, especially when it requires specialized knowledge.

Labeling Functions (LFs), user-created shortcuts that apply labels based on specific patterns or rules, somewhat ease the burden. They allow those who know a subject well to teach the computer how to label similar data. But crafting these functions is tough work and can inadvertently introduce the creator's biases.

Enter WITAN—an algorithm that automates the making of LFs without needing a starting point or guidance. WITAN is especially useful for exploring and sorting data with unknown categories. It reduces the strain and high level of expertise usually needed to make good LFs and even helps avoid human bias.

Even with automation, there's still work for models like Snorkel [1], which take the noisy, sometimes inconsistent labels from LFs and clean them up. They do this by weighing the accuracy of different LFs and harmonizing their labels to train better classifiers.

WITAN is an example of what we call assisted data programming, where we generate large amounts of training data (potentially imperfect) quickly, then refine it into quality data for machine learning. It's a step forward in creating efficient, scalable, and unbiased ways to prepare our data—vital in this era focused on data-driven artificial intelligence.

3 Data preparation

The original paper posits that the WITAN algorithm operates on a given training set $X \in \{0,1\}^{n,m}$, comprising n instances and m features, each encoded in a binary format. This binary representation is a prerequisite for the effective operation of the algorithm's utility function, as well as for assessing the applicability of the resulting labeling functions. Consequently, datasets initially presented in non-binary formats must undergo conversion to binary features.

The research utilizes datasets exemplified by sets of binary bag-of-words features — these discrete features signal the occurrence of specific words within segments of text and are instrumental in the construction of labeling

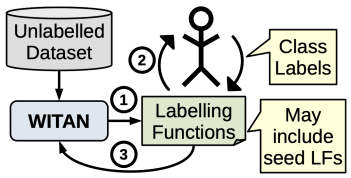

Figure 1. Overview of WITAN, showing ① the initial automatic generation of labeling functions, ② the expert's review process and assigning of classes, and ③ the enhancement of an existing set of LFs.

functions for tasks such as text classification and sentiment analysis. For instance, when analyzing movie reviews, the algorithm might consider the presence or absence of particular terms like "amazing" or "boring" as criteria for evaluating sentiment.

4 The WITAN algorithm

WITAN is an algorithm that discovers useful labeling functions (LFs) without needing any human instructions beforehand. It works with unlabelled data that's already been sorted into binary features and constructs different LFs that cover various aspects of the data (as shown in Figure 1, step 1). A domain expert will then review the LFs that WITAN comes up with and decide (see Figure 1, step 2):

- Which LFs seem appropriate
- What kind of labels these LFs should be applying

4.1 The utility function

4.1.1 Core Concept of the Utility Function

The utility function in WITAN has a crucial role—it selects the LFs that have the best shot at correctly sorting our data. Think of it as a smart filter that finds patterns or rules which reduce guesswork and improve certainty about the data's classification, even when we aren't sure what categories we're working with in the first place.

4.1.2 Dealing with Unknown Class Labels

WITAN cleverly addresses the challenge of working without any predetermined class labels by betting on one key idea: certain patterns in the dataset are likely linked to the unknown classes we want to uncover. By focusing on these indicative patterns, WITAN can discern which features are likely important for sorting data. This approach enables WITAN to select features wisely and guess the roles they play in classification, relying on the hidden structures in the data itself, even without any pre-given labels.

4.1.3 The Utility Function's Calculations and Goals

The utility function within WITAN plays a vital role by evaluating the information gain each feature contributes. It uses a formula $IG\left(x_{*_j}, c^{\lambda}\right) = H\left(x_{*_j}\right) - H\left(x_{*_j} \mid c^{\lambda}\right)$, where H denotes the entropy or the uncertainty level in the data. The function measures how well a feature can clarify the classifications within the dataset, thus determining the effectiveness of each labeling function (LF). WITAN defines this utility function as:

$$U_{X,\bar{H},\gamma,w}(\lambda) = \sum_{j=1, j\notin d^{\lambda}}^{m} w_j \sum_{i=1, c_i^{\lambda}=1}^{n} \max\left(0, \bar{h}_{i,j} - H\left(x_{*j} \mid c^{\lambda}\right)\right)^{\gamma}$$

Key elements of this utility function include:

- **Comprehensive Evaluation:** The function aggregates the information gain from all features and data instances that an LF relates to.
- **Focus on Critical Features:** It sets out to identify LFs based on the most telling features, paving the way for LFs that are likely to predict class categories accurately.
- **Enhancement through Parameters:** By integrating an entropy matrix and adjusting the parameter γ in its formula, WITAN places a higher value on LFs that offer significant predictive insight from fewer instances. This emphasizes the algorithm's preference for precise predictions even if they come from a smaller data sample.

An **entropy matrix**, $\bar{H}$, is a component of the utility function that holds the initial measure of uncertainty for each feature across all instances before any labeling functions are applied. It serves as a benchmark to determine whether an LF can provide additional clarity about the class structure of the data. As WITAN discovers effective LFs and updates their contributions, this matrix adapts, reflecting the reduced uncertainty due to the application of each selected LF. It effectively captures the evolving state of knowledge about the dataset as the algorithm processes different LFs, optimizing for those that are poised to decrease ambiguity the most.

In essence, the utility function's aspiration to minimize uncertainty within the collection of binary data points is fundamental to guiding WITAN's approach. It refines the selection to LFs that can meaningfully and accurately bolster the classification endeavor.

4.2 The core algorithm

4.2.1 Introduction

The WITAN algorithm is built to automatically create a batch of labeling functions (LFs) from plain data with binary features. This automated step converts unorganized data into neatly sorted groups suitable for machine learning applications.

Each labeling function λ crafted by WITAN is capable of assigning a user-determined class label y^{λ} to specific pieces of data it recognizes, known as its 'coverage'. The 'coverage' is essentially a list that the LF follows, tagging data that matches certain qualities, such as mentioning a specific word in a review.

For example, in a batch of movie reviews, WITAN might form an LF that focuses on reviews with the word "amazing". The so-called coverage vector c^{λ} for this function would be the actual list of reviews that fit this description. The role of the user is then to give meaning to this list—maybe deciding that "amazing" reviews should be labeled positive, while others are left without a label.

In summary, WITAN identifies potential LFs that can specify which instances in the dataset they apply to, while not initially assigning any labels. The term 'coverage vector' refers to the set of data instances each LF is relevant to, essentially which pieces of data meet the LF's criteria. WITAN's job is to suggest LFs with broader and more useful coverage. However, it is up to the domain expert to review these suggestions and decide what labels, like 'positive' or 'negative', should be attached based on their own analysis.

4.2.2 Initial Setup

- **Entropy Matrix Initialization:** WITAN initiates its process by preparing an entropy matrix, H, that quantifies the spread of information for each binary feature across the instances in the dataset X. This preparatory measure is vital for calculating the relative potential of different LFs that WITAN will consider.
- **Candidate LFs Creation:** Alongside the entropy matrix, a set of basic candidate LFs is formulated, with each one corresponding to straightforward conditions on single binary features. These foundational LFs serve as precursors for more sophisticated rules that WITAN will later derive.

4.2.3 User Input

- **Minimum Coverage Constraint:** The user has control over the minimum data coverage of LFs with the threshold $C^{\min}$. This parameter ensures that the algorithm prioritizes LFs that affect a significant portion of the dataset, enhancing their practical relevance
- **Seed LFs:** The process also allows for the incorporation of user-provided seed LFs, denoted as $\tilde{\Lambda}$, which contribute to the initial set of selected LFs, Λ that WITAN uses. These seed LFs act as guided starting points for the algorithm's learning journey.

In simpler terms 1

The WITAN algorithm allows users to influence its labeling process by setting rules on how widely an LF needs to apply to the data. Additionally, users can give WITAN initial 'hints' in the form of seed LFs, which are early examples that help steer the algorithm towards more promising labeling patterns.

4.2.4 Main Loop

- **LF Selection:** With every iteration, WITAN selects the LF that most effectively reduces uncertainty—measured by the utility function U—and incorporates it into the set Λ.
- **Entropy Matrix Update:** Upon choosing an LF, WITAN updates its entropy matrix $\bar{H}$. If the LF sheds new light on how to classify instances—which means providing lower entropy or uncertainty values—WITAN adjusts its records accordingly for each feature within such instances. This ensures each LF selection is informed by the latest and most reduced levels of uncertainty.
- **Diversity and Coverage Enhancement:** Through this process, WITAN is adept at continually finding LFs that expand coverage by identifying previously unaddressed instances or by capturing new aspects of the data, thus enhancing the variety and breadth of the LFs.

4.2.5 Stopping Criterion

The process of selecting labeling functions (LFs) within WITAN continues until the collected set of LFs, Λ, accounts for a user-defined minimum portion of the training data, a boundary referred to as $C^{\min}$. Upon meeting this criterion, the algorithm presents its collection of LFs for the domain expert to evaluate and label accordingly.

In simpler terms 2

WITAN keeps generating LFs until enough of the dataset is represented under the guidelines specified by the user.

4.2.6 Additional Procedures

- **Feature Weights and Candidate Updates:** WITAN is designed with flexible mechanisms to adjust the significance or weights of features when calculating utility and to refresh the pool of candidate LFs after each selection round. It also has the capability to enhance an LF with additional criteria before it becomes part of the finalized set, Λ.

- **Extensions:** The framework of WITAN is extensible, allowing for the introduction of modifications that alter the way feature importance is gauged, candidate LFs are renewed, and LFs are expanded upon. This level of adaptability supports WITAN's application across various kinds of data and labeling objectives.

In simpler terms 3

WITAN incorporates stages in its algorithm for fine-tuning the impact of different features considered during LF selection and enhances its collection of candidate LFs as it progresses. This adaptability enables the algorithm to be customized for diverse datasets and labeling requirements.

4.2.7 Key Takeaways

The WITAN algorithm's process of discovering useful labeling functions hinges on several core operations and principles:

- **Iterative and Dynamic Process:** WITAN works through cycles, consistently refining its choice of LFs. With each iteration, it reassesses and enhances its selection based on the updated set of LFs, ensuring a progressive refinement in the labeling strategy.

- **Pursuit of Information Gain:** A central aspect of WITAN's method is its emphasis on selecting LFs that contribute the most significant information gain. By revising the entropy matrix regularly, WITAN ensures that the most informative and varied aspects of the dataset are captured.

- **User-Guided Flexibility:** WITAN is designed to be malleable to the user's needs. It allows for user inputs at various stages, enabling it to be tailored to different dataset types and labeling challenges, molding the algorithm to the specifics of the task at hand.

In simpler terms 4

WITAN's strength lies in its iterative refinement, its focus on selecting the most informative labeling functions, and its ability to adapt to user guidance, making it versatile for a range of data scenarios.

5 WITAN in action

5.1 Demonstrating WITAN with a Simple Example

Suppose we have a small dataset of customer reviews:

- Review 1: "Love this product, amazing quality!"
- Review 2: "Terrible experience, very disappointed."
- Review 3: "Exceptionally good, highly recommend."
- Review 4: "Poor quality, not worth the price."

We can illustrate how WITAN would process this example through a process called feature extraction.

5.1.1 Feature Extraction

For the sake of illustration, let's transform the sentiment expressed in these reviews into binary features. We select a few positive and negative keywords to demonstrate:

- Feature 1: Indicates the presence of positive words like "love" or "amazing."
- Feature 2: Indicates the presence of negative words like "terrible" or "poor."
- Feature 3: Indicates the presence of positive endorsements like "recommend" or "good."
- Feature 4: Indicates the presence of expressions of disappointment like "disappointed" or "not worth."

The table below showcases our dataset, now encoded with binary features:

Review	Feature 1	Feature 2	Feature 3	Feature 4
1	1	0	0	0
2	0	1	0	1
3	0	0	1	0
4	0	1	0	1

Table 1. Dataset encoded with binary features representing the presence (1) or absence (0) of specific sentiment-associated keywords.

5.1.2 Initial Labeling Functions (LFs)

WITAN begins by creating a starting set of labeling functions (LFs) based on the identified features. For example:

- LF1: Assign a "Positive" label if Feature 1 (words like "love" or "amazing") appears in the review.
- LF2: Assign a "Negative" label if Feature 2 (words like "terrible" or "poor") is present.

...and so on, with additional LFs tied to the other extracted features.

5.1.3 Selecting and Refining LFs

Armed with its utility function, WITAN assesses each LF for its ability to add clarity to the data's classification, prioritizing those that provide valuable information. As the process iterates, WITAN might adjust these LFs—combining some, refining others, or incorporating fresh user input.

5.1.4 Classification Process

The selected LFs are then applied to classify the reviews, with their output being something like:

- Review 1 is labeled "Positive" due to LF1.
- Review 2 gets a "Negative" label, triggered by LF2 and LF4.
- Review 3 is considered "Positive" thanks to LF3.
- Review 4 is deemed "Negative", indicated by LF2 and LF4.

5.1.5 User Feedback and Iteration

At this stage, a user reviews WITAN's work, confirming accurate labels or pointing out corrections. This feedback becomes part of WITAN's learning cycle, allowing the algorithm to fine-tune its LFs and enhance the accuracy of future classifications.

5.1.6 Further Refinement and Expansion

After receiving user feedback that Review 3, labeled "Positive", contains nuanced sentiment not captured by the initial LFs, WITAN might introduce:

- An extended LF that combines Features 1 and 3 to better capture the expression of strong positive sentiment.
- A mechanism to handle mixed expressions within reviews, such as "not only good but also amazing", reinforcing the positive classification.

Subsequently, WITAN may propose more advanced LFs like:

- LF5: If a review contains combinations of Features 1 and 3, it might indicate a stronger positive sentiment and be labeled "Highly Positive".

This iterative process allows WITAN to refine its understanding over time, attaining higher accuracy and nuance in the labels it suggests and applies.

5.2 Review of WITAN's Performance

5.2.1 Binary Classification Results

In an evaluation of binary classification tasks:

- **Performance With Limited Interactions:** WITAN shows a strong performance when user interactions are limited to 25, demonstrating its capability to deliver valuable results swiftly with minimal user input.
- **Comparing Witan Variants:** At higher levels of user interaction (100 interactions), the stripped-down version, Witan-Core, demonstrates a better performance than the standard Witan. This suggests Witan-Core may efficiently refine its predictions with increasing user input.
- **Comparative Supervised Performance:** The close approximation of Witan-Core's results to those of fully supervised methods highlights its utility in environments where access to labeled data is challenging.

5.2.2 Insights from Binary Classification

From the results of binary classification analysis, several key findings emerge:

- **User Investment versus Output:** The trend shows that Witan's efficiency is more pronounced with lesser user effort, implying a decrease in benefits with increasing interaction numbers.
- **Comparison to Alternative Methods:** When compared to other labeling techniques such as active learning, Witan stands out - particularly when user interaction is limited, indicating its strength in learning from broad labeling patterns rather than specific examples.
- **Knowledge Representation:** The labeling functions generated by Witan not only aid classification tasks but also serve as an understandable and modular form of capturing domain knowledge, surpassing individual instance labels in maintainability.

5.2.3 Multi-Class Classification Results

Assessments in multi-class classification scenarios reveal:

- **Dynamic Shifts:** The dynamics of performance favor Witan-Core over Witan even at lesser interaction counts, evidencing a shift compared to binary classification results.
- **Core Advantages:** Multi-class contexts, particularly those with narrower class distributions, seem to benefit from the capacity adjustments inherent in Witan-Core.

5.2.4 General Observations

From a broader perspective, Witan exemplifies:

- **Interactive Efficiency:** Witan is efficient for interactive use, delivering significant results within limited user involvement time frames.
- **Versatility:** The algorithm exhibits versatility, adapting effectively across a diverse range of datasets and classification challenges.
- **Performance Trends:** Though all methods tend to reach a similar performance level eventually, Witan notably achieves near-optimal outcomes more expediently.

5.2.5 Critical Analysis

The in-depth examination of Witan acknowledges:

- **Identified Constraints:** There are scenarios where Witan's efficacy may decline, particularly if new LFs incorporated later are of inferior quality or they introduce imbalance across the class labels.
- **The Quality-Quantity Equilibrium:** The experimental findings underscore that optimal results depend on a judicious balance between the number of LFs created and the precision they can offer—the quantity must not outstrip the quality.

5.2.6 Ethical Considerations and Biases

While heuristic labeling functions can be algorithmically generated with tools like WITAN, they are not immune to biases affecting label accuracy and trustworthiness. Biases can especially proliferate when using crowdsourced or large unvetted datasets, leading to unreliable and prejudiced models [2]. It's crucial that any initial user-defined rules are critically assessed and adhere to rigorous standards to mitigate these biases. Inattention to this may result in the persistence and magnification of any bias present in the user's guidelines, particularly when relying on expert heuristics drawn from unrepresentative datasets.

However, the transition to automated label generation presents a positive opportunity to improve ethical labor practices. Labeling, currently a task often outsourced to underpaid workers in the Global South, can be remodeled to rely less on controversial labor practices, providing an ethical boost to the data processing industry [3].

Both the challenges and potentials of automated data labeling need to be taken into account when considering the broader implications of this research.

5.3 Deployment and Execution

The implementation of the WITAN algorithm, along with the experiments conducted, is made accessible through a GitHub repository. For ease of use and widespread deployment, the code is encapsulated within a Docker container, streamlining the process of running the algorithm.

To demonstrate WITAN in a real-world context, the algorithm was applied to the IMDB review dataset, which consists of reviews categorized as either negative or positive. An automated simulation, replacing an actual user, was devised to evaluate the algorithm's performance. This simulation involves endorsing labeling functions (LFs) that envision at least a 20% better accuracy rate than what would be achieved by guessing at random.

For a more extensive analysis, the algorithm was allowed to iteratively generate LFs beyond the scope of the examples presented in the original paper. This extended run sheds light on the algorithm's capacity to continually evolve its labeling functions and decision-making strategy.

6 Conclusion

Throughout our evaluation, Witan has emerged as a transformative tool in the domain of data programming—especially potent in settings where labeled data is scarce. Its exceptional method for autonomously generating labeling functions, without reliance on preset class labels, directly confronts a notable obstacle within the machine learning community.

The algorithm's utility function leverages insights from rule-learning algorithms to adeptly traverse the intricacies of unlabeled datasets. This innovation greatly diminishes the dependency on manual labeling, thereby expediting the classification process.

Our empirical testing showcases Witan's competencies across both binary and multi-class classification tasks, drawing attention to its ability to attain remarkable levels of accuracy with minimal user input. Such effectiveness

in managing unlabeled data, along with its limited requirement for expert intervention, heralds Witan as a forward leap in the quest to streamline machine learning practices.

In closing, Witan's strategy for the unsupervised generation of labeling functions stands as a significant contribution to the field of data programming. It presents a viable, avant-garde solution to the challenges of data classification, enhancing the practicality and productivity of machine learning initiatives.

Acknowledgments

We would like to acknowledge the authors of the WITAN algorithm for their groundbreaking work in the domain of data programming. The original research paper, poster presentation, and publicly available source code provide valuable insights and resources that have greatly assisted our review process.

Original paper: *Witan: Unsupervised Labelling Function Generation for Assisted Data Programming* [4].

Poster presentation: Available online at this link [5].

Source code: Accessible on GitHub at this link [6].

References

[1] Alexander Ratner, Stephen H. Bach, Henry Ehrenberg, Jason Fries, Sen Wu, and Christopher Ré. Snorkel: Rapid training data creation with weak supervision. *arXiv preprint arXiv:1711.10160*, 2017. doi:10.48550/arXiv.1711.10160. URL https://doi.org/10.48550/arXiv.1711.10160.

[2] Yan Li, Maria De-Arteaga, and Maytal Saar-Tsechansky. When more data lead us astray: Active data acquisition in the presence of label bias. In *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing*, volume 10, pages 133–146, 2022. doi:10.1609/hcomp.v10i1.21994. URL https://doi.org/10.1609/hcomp.v10i1.21994.

[3] Madhumita Murgia. Ai's new workforce: the data-labelling industry spreads globally. *Financial Times*, 7 2019. URL https://www.ft.com/content/56dde36c-aa40-11e9-984c-fac8325aaa04.

[4] Benjamin Denham, Edmund M-K. Lai, Roopak Sinha, and M. Asif Naeem. Witan: Unsupervised labelling function generation for assisted data programming. *PVLDB*, 15(11):2334–2347, 2022. doi:10.14778/3551793.3551797.

[5] Benjamin Denham, Edmund M.-K. Lai, Roopak Sinha, and M. Asif Naeem. Witan: Unsupervised labelling function generation for assisted data programming. https://ben-denham.github.io/witan/witan-poster.pdf.

[6] Witan github repository. https://github.com/ben-denham/witan, 2023.