Identifying High Value Opportunities for Human in the Loop Lexicon Expansion

Alfredo Alba
IBM Research Almaden, CA, US
aalba@us.ibm.com

Chad DeLuca
IBM Research Almaden, CA, US
delucac@us.ibm.com

Anna Lisa Gentile
IBM Research Almaden, CA, US
annalisa.gentile@ibm.com

Daniel Gruhl
IBM Research Almaden, CA, US
dgruhl@us.ibm.com

Linda Kato
IBM Research Almaden, CA, US
kato@us.ibm.com

Chris Kau
IBM Research Almaden, CA, US
ckau@us.ibm.com

Petar Ristoski
IBM Research Almaden, CA, US
petar.ristoski@ibm.com

Steve Welch
IBM Research Almaden, CA, US
welchs@us.ibm.com

ABSTRACT

Many real world analytics problems examine multiple entities or classes that may appear in a corpus. For example, in a customer satisfaction survey analysis there are over 60 categories of (somewhat overlapping) concerns. Each of these is backed by a lexicon of terminology associated with the concern (e.g., “Easy, user friendly process" or "Process confusing, too many handoffs”). These categories need to be expanded by a subject matter expert as the terminology is not always straight forward (e.g., “handoffs” may also include “ping-pong” and “hot potato” as relevant terms).

But given that Subject Matter Expert time is costly, which of the 60+ lexicons should we expand first? We propose a metric for evaluating an existing set of lexicons and providing guidance on which are likely to benefit most from human-in-the-loop expansion. Using our ranking results we achieved $\approx 4\times$ improvement in impact when expanding the first few lexicons off our suggested list as compared to a random selection.

ACM Reference Format:
Alfredo Alba, Chad DeLuca, Anna Lisa Gentile, Daniel Gruhl, Linda Kato, Chris Kau, Petar Ristoski, and Steve Welch. 2019. Identifying High Value Opportunities for Human in the Loop Lexicon Expansion. In *HumBL2019. The third international workshop on Augmenting Intelligence with Bias-Aware Humans-in-the–Loop. In the Web Conference 2019 Companion volume.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3308560.3317305

1 INTRODUCTION

The data in real world text analytics problems is messy. It typically contains a plethora of categories with varying degrees of relatedness towards the task at hand. Each category can be represented by a lexicon; containing a representation, as exhaustive as possible, language variants authors use to express an entity belonging to the given category. With the proper set of lexicons that represent the necessary categories, further analysis can be conducted, such as survey classification, semantics, category relationship analysis, etc. These analyses extract the insights that can ultimately enable a data driven business decision making process.

We illustrate this with a case study of customer service satisfaction surveys. These surveys are typically designed to collect data to answer specific business questions, e.g. to understand customers' concerns about specific categories of issues. More than one category may appear in a single survey; e.g., a customer can be satisfied with the service agent's empathy towards them while still being upset about the amount of time it took to solve the issue. In order to derive the desired insights from the data, it is essential to identify and effectively extract the meaningful category identifiers.

As these are surveys written in the customer's own language, the various phrasings, misspellings, etc. need to be interpreted and well understood. Acquiring a deep understanding of the language a specific community uses to describe many categories of issues - in this specific case more than 60 - at any given point in time is a considerable challenge. A brute force approach for this task can be effective, but extremely time consuming. Additionally, asking a Subject Matter Expert (SME) to quickly score tens of thousands of documents becomes an daunting task as review fatigue sets in.

In our approach, the SMEs quickly curate the lexicons for each category from their experience and they manually inspect as many survey examples as possible. While the manual approach can be effective, it provides only a point in time solution. Over time, the language of a community drifts due to both internal changes (e.g., call centers moving, new product releases, etc.) as well as external changes in the common usage (slower but still of importance over decades of data). Thus, lexicon curation becomes a continuous improvement exercise to maintain it's relevancy.

Humans, even SMEs, are not perfect in their performance, as demonstrated by the number of human reliability assessment techniques [15]. The factors that impact the human reliability are numerous [16, 25]. These factors have a considerable impact on lexicons solely curated by an SME using a manual process. Therefore the higher the target quality of the manually curated data, the higher the cost (1).

On the other hand, traditional distantly or closely supervised approaches require considerable up front investments in developing

HumBL2019, May 2019, San Francisco, CA

ACM ISBN 978-1-4503-6675-5/19/05.
https://doi.org/10.1145/3308560.3317305

Figure 1: In real world analytics applications, the Subject Matter Expert is often confronted with dozens of semantic assets such as lexicons that must be refreshed. Without guidance, they can end up spending time working on lexicons whose improvement has very little impact on the overall business objectives.

large amounts of high quality representative training data; and it is subject to the same decay over time as highlighted above.

This might lead one to consider an unsupervised approach. However with these the cost can be even higher due to all the data scientist time that has to be spent on feature engineering, which is indispensable in assuring there are no biases or poisoned features, potentially rendering the resulting model unusable [5, 6]. Although recent work [4] has taken aim at addressing this issue via adversarial networks, among other approaches, depending on the target model this remains a very active area of research without a definitive solution.

An alternative is a human-in-the-loop system where the human is assisted by a data driven cognitive system in the task of lexicon curation. The system helps by providing candidates for human adjudication. As the candidates are adjudicated the system learns the semantics of the categories at hand and rapidly produces higher quality candidates for the SME to adjudicate.

This human-in-the-loop system maximizes the impact of the SME's time and effort. By putting the greatest value on the human's time, resources usage is optimized in a way that reflects real world requirements. Business value is achieved by identifying the most efficient path, given available knowledge and resources.

The system provides the added benefit of keeping the focus on areas of rapid change. It might be common for lexicons in most categories to rarely require modification, however some categories undergoing rapid change will surface more frequently, thus exposing themselves as more dynamic.

We propose a novel technique to proactively identify which lexicons can benefit most from expansion. This can be used both to order the expansion efforts, as well as (using a threshold) monitor a collection of lexicons and alert when expansion is warranted.

The major contribution of this paper is a set of novel features and way of combining them which can be used to detect when expansion is likely to be beneficial. We discovered that the most important features to take into account are: the size of the initial lexicon, the number of initial hits, the lexicon intra-similarity and the confidence of the used annotation model.

2 STATE OF THE ART

Lexicons, ontologies, and linguistic resources are the backbone of many NLP and information retrieval systems and applications. The automatic construction of such resources has been the focus of research for many years. Riloff and Jones [20] propose one of the first approaches to extract dictionaries from unstructured text. Starting from a few seed terms, it learns extraction patterns, which are then used to expand the seeds and iteratively repeat the process. In the following years, a number of similar approaches have been developed [3, 7, 14, 21]. However, all these approaches require NLP parsing for feature extraction, and thus have a reliance on syntactic information for identifying quality patterns. Hence, such approaches underperform on not-so-well structured or or grammatically incorrect text content, such as user-generated text. Furthermore, completely automatic iterative methods without human-in-the-loop can easily generate semantic drift - a few spurious extraction patterns can exponentially increase the inclusion of incorrect items in the lexicon.

To address these issues, recent works propose feature-antagonistic approaches for dictionary generation [2, 12]. These approaches couple deep neural language models with tight human supervision to assist the user in building and maintaining domain-specific dictionaries. While these approaches are able to efficiently and effectively extend lexicons, in cases where the number of input lexicons is high, they do not provide guidance on which lexicons are likely to benefit most from human-in-the-loop expansion.

While there are no approaches in the literature that address this problem directly, there are many approaches that, given a set of items, can be used for ranking and selecting the most promising item that will lead to the highest reward. For example, active learning tries to identify the most important data instances to be labeled by a human oracle, i.e., identifies the data points that would improve the model performance the most. In the literature there are many ranking and selection strategies proposed [24], e.g., uncertainty sampling [18], density weighted uncertainty sampling [10, 19], diversity [8], QUIRE [13] and Bayesian methods such as BALD (Bayesian Active Learning by Disagreement) [11].

Another type of approaches that focus on the problem of ranking and selecting the most beneficial items, are bandit approaches [22, 23]. More precisely, bandit approaches develop a set of strategies that given a set of choices/items are able to rank the probability for selecting each of them in a way that maximizes the expected gain. This problem has been studied for many years resulting in a plethora of existing approaches [17].

While there are many approaches for document ranking and selection in machine learning, to the best of our knowledge, no existing approach addresses the challenges of ranking lexicons and providing guidance on which are likely to benefit most from human-in-the-loop expansion.

Table 1: Surveys corpus and lexicons statistics. Average (avg.), minimum (min), maximum (max) and median (med.) values for the lexicons size, number of Total Tagged Documents (TTD), and intra-lexicon similarity, in the original lexicons (Original) and after the extension of the lexicons (Extended)

	#Surveys	#Lex.	Lexicon Size				#TTD					Intra-Lexicon Similarity			
			avg.	min	max	med	total	avg.	min	max	med.	avg.	min	max	med
Original	378,000	65	18.00	2	96	8	147,714	5,908.56	33	44,925	1,252	0.39	0.13	0.86	0.39
Extended	378,000	65	51.96	5	125	45	203,839	8,153.56	73	48,965	2,543	0.35	0.24	0.52	0.34

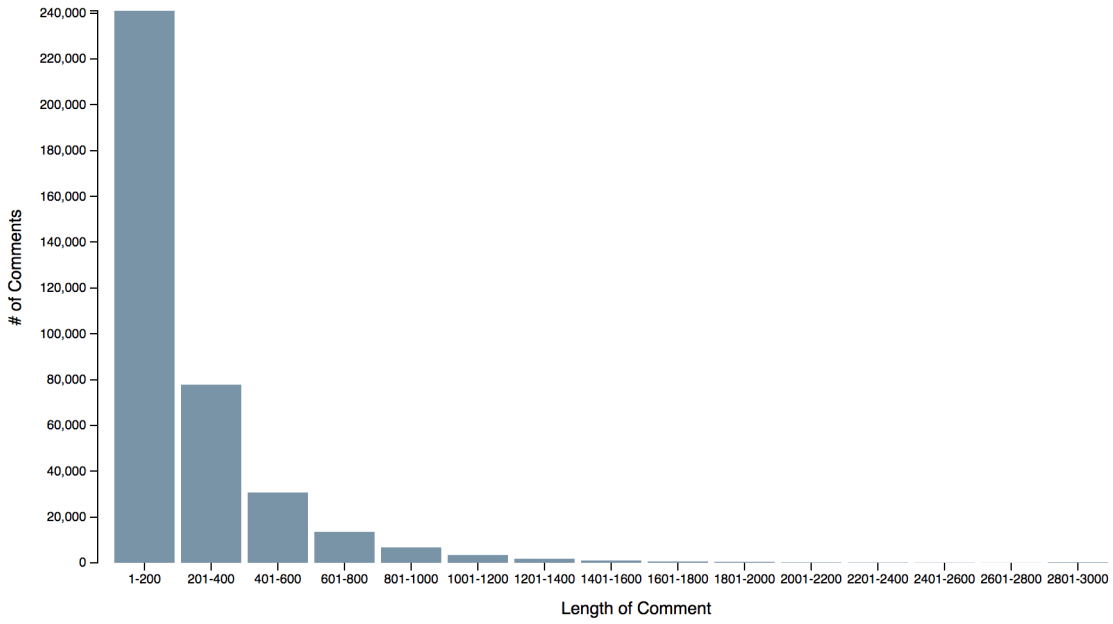

Figure 2: Customer Satisfaction surveys contain a free text field as well as structured questions. The free text responses tend to be very short, but there are cases of respondents including upwards of a page of text detailing their responses on their interactions with the call center.

3 APPROACH

The measure we are seeking to optimize is the count of "Total Tagged Documents" (*TTD*) in the corpus we are examining. Thus, if adding the phrase "great customer service" resulted in 2,900 more documents being tagged (as they each had this phrase appearing), we would say that adding that phrase had a "score" of +2, 900.

Phrased as such, we have a classic prediction problem ahead of us. We wish to produce an ordered list of lexicons to expand, such that the first lexicon we suggest is most likely to result in the highest increase in *TTD*. We score ourselves using the Spearman rank correlation metric for list ordering.

There are then two steps; identifying (and developing) features that correlate meaningfully with *TTD*, followed by identifying an effective way of combining those features to create a useful ordering.

3.1 Features

Features available to us are those that can be measured on the a priori lexicons before any human involvement. We note that in the case of "continual improvement" situations, there may be more data available, but we leave that for future work.

Several features were found to be useful. Two basic ones were number of terms in the lexicon, and the *TTD* of the lexicon initially. The intuition here is that large lexicons are perhaps more fully expanded already, and that lexicons that cover a large number of documents are focusing on concepts that are prevalent in the source corpus.

Another feature was obtained via running a `word2vec` style expander (Explore and Exploit [1, 12]) and looking at how "close" in vector space the terms in the initial lexicon were. This gives a sense of how "tight" the initial terminology was. We expect a more fully expanded lexicon to be more "diffused" (having less coherence) as more "edge concepts" are included. The feature is calculated as the average cosine similarity of all the terms in the lexicons (the higher the value, the more similar are the terms inside the lexicon).

A fourth feature type was obtained by running an initial Explore and Exploit expansion to look at the curve of confidence and cumulative confidence per term for the first 20 terms. This aims to be

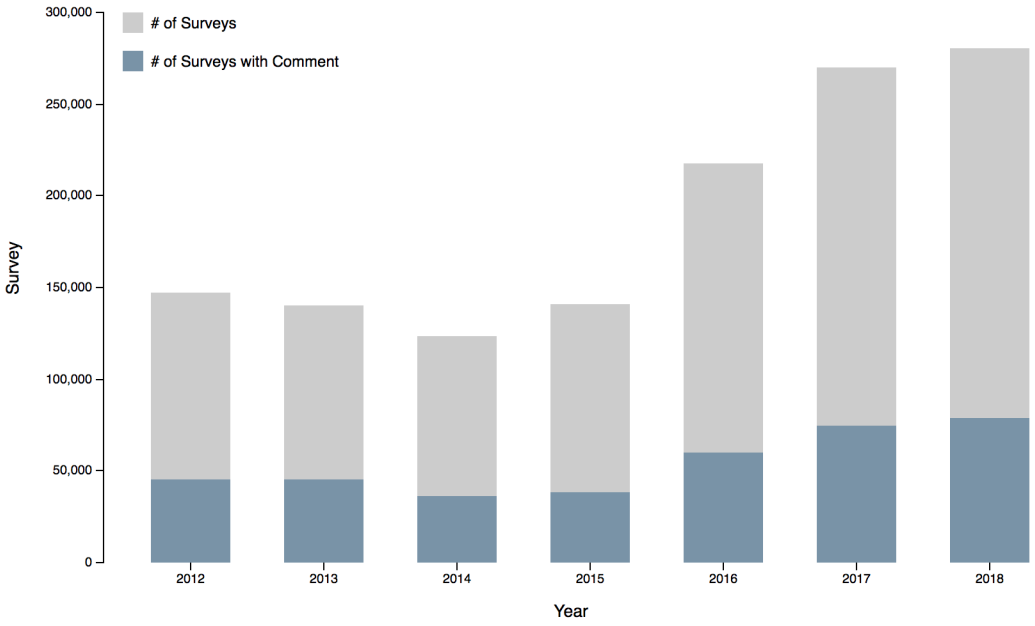

Figure 3: Surveys span 7 years. Even though most do not contain a free text comment, around 50,000 a year do – many more than it would be practical to human review for each of 60+ categories.

a measure of the "low hanging fruit" of the algorithm. While it is not exact (as there has been no human involvement, nor is it very much like the Glimpse[9] engine which is also used by the human) it does prove to be one of the better single features.

Our fifth feature looks at the number of hits a lexicon gets each year for the 7 years we have available in the survey dataset. We normalize by the total number of surveys in the year and then fit a line to the results. This "slope" gives us a measure of how quickly the initial terms are decreasing in prevalence, with the notion this can be a measure of how quickly the terms are going "stale" as language evolves around the entity of interest.[1]

3.2 Regression Models

We approach the problem of ranking lexicons as a regression problem, i.e., we try to predict the increase in *TTD* after the lexicon expansion, which is then used to rank the lexicons. To effectively combine the extracted features, we use 4 standard regression models: (i) Linear Regression, (ii) K-Nearest Neighbors, (iii) Decision Trees and (iv) Neural Network.

4 EVALUATION

4.1 Dataset

The dataset used for this experiment consists of customer survey results spanning 7 years. The survey asks the customer to rate their experience in a number of categories using a numerical scale.

[1]Note: While this applies only to the single dataset we tested on, the approach is transferable to any "tune up" of a domain. This is especially true when you are doing year-to-year tune-ups and have data to fit a predictor to, based on what worked well in the last round of tune-ups.

Table 2: Spearman's rank correlation ρ for lexicon rank prediction, using 4 models: Linear Regression (LR), K-Nearest Neighbors (KNN), Neural Network (NN) and Decision Tree (DT)

Model	ρ
LR	0.32
KNN	0.71
NN	0.41
DT	**0.86**

There is also an optional unstructured text field for the customer to add any additional comments. All questions are optional, so many questions are left unanswered. The dataset contains survey results from 2102 to 2018. There are a total of 132 million surveys in the dataset, about 378K of which contain free form text comments. The length of the comments typically varies between a single word and a couple of paragraphs of text. The distribution of the length of the comments is shown in Figure 2. Figure 3 shows the distribution of comments per year. We can observe that there is a slight increase of comments in the last 3 years. We use the set of 378K surveys to build the Explore and Exploit model [2], which is used for extending the lexicons. We used a set of 65 lexicons which were expanded by a group of 8 Subject Matter Experts. The statistics of the used lexicons before and after the expansion are shown in Table 1. Given an input text corpus and a set of seed examples, the Explore and Exploit approach first builds a neural language model on the input text corpus, and then runs in two phases to identify new potential lexicon entries. The Explore phase tries to identify similar instances

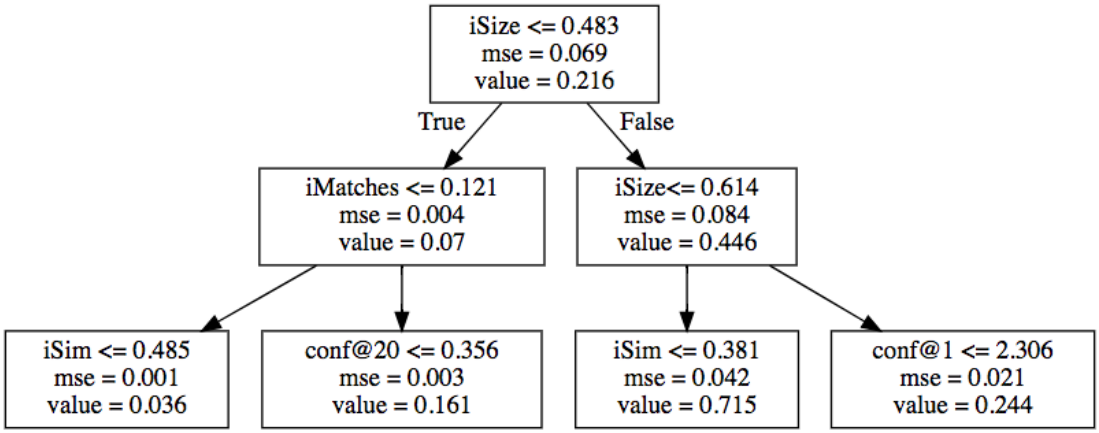

Figure 4: Decision tree model visualization.

to the dictionary entries that are present in the input text corpus, using term vectors from the neural language model to calculate a similarity score. The Exploit phase tries to construct more complex multi-term phrases based on the instances already in the input dictionary. The SME is closely involved in the processed to avoid semantic drift and incorrect entries in the lexicon.

4.2 Results

We consider 4 regression models:

- Linear Regression (LR)
- K-Nearest Neighbors (KNN), with K=3
- Neural Network (NN) with one input layer, three fully connected hidden layers with ReLU activation function, with 10, 8 and 6 neurons respectively, and an output layer with linear activation function.
- Decision Tree (DT), using mean squared error for node splitting criteria and with a maximum tree depth of 5.

For the implementation of the models we use the Python *sklearn* library.

To evaluate the models, we use the Spearman's rank correlation as an evaluation metric, i.e., we evaluate the correlation of the lexicon rank produced by the models and the real rank of the lexicons.

The results are show in Table 2. The Spearman's rank correlation is highest when using a Decision Tree model. Closer analysis of the decision tree structure reveals that the most important features are the size of the initial lexicon (iSize), the number of initial hits (iMatches), the lexicon intra-similarity (iSim) and the model confidence at 1 (conf@1) and at 20 (conf@20). This means that the smallest lexicons with high coherency (e.g. the variety of terms is very low) need to be extended first. The top 3 layers of the decision tree model are shown in Figure 4. We can see that the neural network model is not able to learn a good representation of the data, as the dataset is rather small. The K-Nearest Neighbors model performs quite well with both sets of features, while the Linear Regression could not learn a good model.

We compare the final model to a random selection from the lexicons to expand by calculating *TDD* 3 and 5. Compared to a random selection at 3, we see 406% uplift in *TDD*, and at 5 we still see 387% uplift. This represents a significant time savings/increase in impact for the human Subject Matter Experts (Figure 5).

5 CONCLUSIONS AND FUTURE WORK

In any human-in-the-loop task, human time is by far the most constrained resource a system has to work with. As human-in-the-loop rolls out to more niche domains, there is a limit to how long human involvement can be justified - hence the importance of making the most out of it.

For large analytics projects with dozens of lexicons, keeping them up to date is a time consuming but necessary task. Our approach meshes well with the real world "quarterly update" notion of lexicon maintenance – by identifying where human involvement will make the most difference.

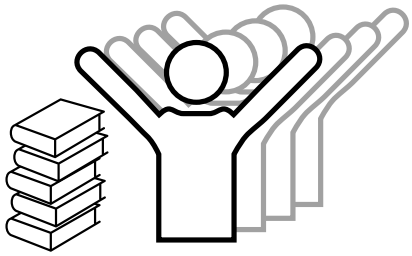

Figure 5: By providing Subject Matter Experts with an ordered list of lexicons, they can start with the most impactful first. The head of this list is 4× more impactful than a random selection.

By allowing the human to identify the lexicons that are most likely to benefit from expansion, time can be planned, experts marshaled, and improvements made in an incremental manner, with the very high likelihood ($\rho = 0.86$) that their work will make the most difference. Overall, this results in a 4× increase in impact for human Subject Matter Experts who are expanding just a few lexicons.

While this approach has worked well for this case, it would be good to test the approach on a wider variety of lexicon sets. One

of especial interest is the Wikipedia/DBpedia categories - can our approach identify new entities to tag into a category, or perhaps even new articles that should be authored?

Additionally, in an incremental setting (e.g., identify a good lexicon to expand, expand it, repeat), taking advantage of the accuracy of the prediction to dynamically adjust the splits in the tree may help better identify the next lexicon to enrich.

REFERENCES

[1] Alfredo Alba, Anni Coden, Anna Lisa Gentile, Daniel Gruhl, Petar Ristoski, and Steve Welch. 2017. Multi-lingual Concept Extraction with Linked Data and Human-in-the-Loop. In *Proc. of the Knowledge Capture Conference*. ACM, 24.

[2] Alfredo Alba, Daniel Gruhl, Petar Ristoski, and Steve Welch. 2018. Interactive Dictionary Expansion using Neural Language Models. (2018).

[3] Rie K Ando. 2004. *Semantic lexicon construction: Learning from unlabeled data via spectral analysis*. Technical Report. IBM THOMAS J WATSON RESEARCH CENTER YORKTOWN HEIGHTS NY.

[4] Nathalie Baracaldo, Bryant Chen, Heiko Ludwig, and Jaehoon Amir Safavi. 2017. Mitigating Poisoning Attacks on Machine Learning Models: A Data Provenance Based Approach. In *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security*. ACM, 103–110.

[5] Marco Barreno, Blaine Nelson, Russell Sears, Anthony D Joseph, and J Doug Tygar. 2006. Can machine learning be secure?. In *Proceedings of the 2006 ACM Symposium on Information, computer and communications security*. ACM, 16–25.

[6] Battista Biggio, Blaine Nelson, and Pavel Laskov. 2012. Poisoning attacks against support vector machines. *arXiv preprint arXiv:1206.6389* (2012).

[7] Sebastian Blohm and Philipp Cimiano. 2007. Using the Web to Reduce Data Sparseness in Pattern-Based Information Extraction. In *PKDD 2007*. Springer, 18–29. DOI:http://dx.doi.org/10.1007/978-3-540-74976-9_6

[8] Klaus Brinker. 2003. Incorporating diversity in active learning with support vector machines. In *Proceedings of the 20th international conference on machine learning (ICML-03)*.

[9] Anni Coden, Daniel Gruhl, Neal Lewis, Michael Tanenblatt, and Joe Terdiman. 2012. SPOT the drug! An unsupervised pattern matching method to extract drug names from very large clinical corpora. *Proc. - 2012 IEEE 2nd Conference on Healthcare Informatics, Imaging and Systems Biology, HISB 2012* (2012), 33–39.

[10] Pinar Donmez, Jaime Carbonell, and Paul Bennett. 2007. Dual strategy active learning. In *ECML*. Springer.

[11] Yarin Gal, Riashat Islam, and Zoubin Ghahramani. 2017. Deep Bayesian Active Learning with Image Data. In *ICML*.

[12] Anna Lisa Gentile, Daniel Gruhl, Petar Ristoski, and Steve Welch. 2019. Explore and Exploit. Dictionary Expansion with Human-in-the-Loop. In *European Semantic Web Conference (TO APPEAR)*. Springer International Publishing.

[13] Sheng-Jun Huang, Rong Jin, and Zhi-Hua Zhou. 2010. Active learning by querying informative and representative examples. In *NIPS*.

[14] Sean P Igo and Ellen Riloff. 2009. Corpus-based semantic lexicon induction with web-based corroboration. In *Proc. of the Workshop on Unsupervised and Minimally Supervised Learning of Lexical Semantics*. Association for Computational Linguistics, 18–26.

[15] Barry Kirwan. 1992. Human error identification in human reliability assessment. Part 2: detailed comparison of techniques. *Applied ergonomics* 23, 6 (1992), 371–381.

[16] Barry Kirwan. 1996. The validation of three Human Reliability Quantification techniquesÃćâĆňâĂİTHERP, HEART and JHEDI: Part 1ÃćâĆňâĂİtechnique descriptions and validation issues. *Applied ergonomics* 27, 6 (1996), 359–373.

[17] Volodymyr Kuleshov and Doina Precup. 2014. Algorithms for multi-armed bandit problems. *arXiv preprint arXiv:1402.6028* (2014).

[18] David D Lewis and Jason Catlett. 1994. Heterogeneous uncertainty sampling for supervised learning. In *ICML*.

[19] Hieu T Nguyen and Arnold Smeulders. 2004. Active learning using pre-clustering. In *ICML*. ACM.

[20] Ellen Riloff, Rosie Jones, and others. 1999. Learning dictionaries for information extraction by multi-level bootstrapping. In *AAAI/IAAI*. 474–479.

[21] Ellen Riloff, Janyce Wiebe, and Theresa Wilson. 2003. Learning Subjective Nouns Using Extraction Pattern Bootstrapping. In *Proc. of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003 - Volume 4*. 25–32. DOI:http://dx.doi.org/10.3115/1119176.1119180

[22] Herbert Robbins. 1985. Some aspects of the sequential design of experiments. In *Herbert Robbins Selected Papers*. Springer, 169–177.

[23] Gunter Rudolph. 1997. Reflections on bandit problems and selection methods in uncertain environments. In *International Conference on Genetic Algorithms*. Citeseer.

[24] Burr Settles. 2012. Active learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning* 6, 1 (2012), 1–114.

[25] JC Williams. 1988. A data-based method for assessing and reducing human error to improve operational performance. In *Human Factors and Power Plants, 1988., Conference Record for 1988 IEEE Fourth Conference on*. IEEE, 436–450.