# Building Effective Query Classifiers: A Case Study in Self-harm Intent Detection

Ashiqur R. KhudaBukhsh[1]
Carnegie Mellon University
Pittsburgh, PA 15213
akhudabu@cs.cmu.edu

Paul N. Bennett
Microsoft Research
Redmond, WA 98052
pauben@microsoft.com

Ryen W. White
Microsoft Research
Redmond, WA 98052
ryenw@microsoft.com

## ABSTRACT

Query-based triggers play a crucial role in modern search systems, e.g., in deciding when to display direct answers on result pages. We address a common scenario in designing such triggers for real-world settings where positives are rare and search providers possess only a small seed set of positive examples to learn query classification models. We choose the critical domain of self-harm intent detection to demonstrate how such small seed sets can be expanded to create meaningful training data with a sizable fraction of positives examples. Our results show that with our method, substantially more positive queries can be found compared to plain random sampling. Additionally, we explored the effectiveness of traditional active learning approaches on classification performance and found that maximum uncertainty performs the best among several other techniques that we considered.

## Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*query formulation, search process*.

## Keywords

Query classification; Self-harm; Active learning.

## 1. INTRODUCTION

Query-based triggers play an important role in modern information retrieval (IR) systems. Such triggers can be used to decide when to display rich direct answers (weather or stocks) on search engine result pages (SERPs), target display advertisements to particular queries or query classes, or issue other specific notifications to searchers. While a high-accuracy query classifier can help such systems reach all interested searchers, misclassification has a cost of showing notifications or advertisements in inappropriate contexts which may annoy or frustrate searchers. In the construction of query classifiers, search providers often only possess a small *seed set* of target queries, which can be insufficient for training. Positive queries (examples of queries that should be targeted) are rare in many query classification tasks. A common need is to expand the seed set to identify more positives without adding many negatives.

In this work, we choose one particularly important query classification problem, self-harm intent detection, as a case study to highlight the challenges and approaches to building an effective query classifier for a targeted domain. Currently, the major Web search engines respond to queries such as [how to kill yourself] with an answer-like treatment that provides a telephone number for the National Suicide Prevention Lifeline. The research challenge in this domain is to build classification models that can automatically and accurately detect when a query indicates an intent of the searcher for self-harm. To address that challenge, we make the simplifying assumption that the classifier solely relies on the text of query statements for training and prediction, and does not use additional information such as result clicks, cursor movements, and part of speech tagging—all of which have been used for query classification purposes [6][8][9]. This has the practical benefit that classifying solely based on query features reduces latency which is crucial for Web-scale use.

We make the following contributions with our research. We examine different notions of relatedness in query classification, and whether they provide different benefits when expanding a small set of seed queries to a much larger training and evaluation set. Additionally, this is the first work in IR on self-harm, a critically important topic. Finally, we believe that our work will stimulate discussion in the research community on this important, albeit sensitive, issue with specific questions such as when any intervention should be triggered, how intervention can be best provided, how specific intervention assistance should be, whether query re-writing should occur on such queries and the general question of how search engines should operate in such sensitive areas.

## 2. RELATED WORK

Query intent classification has been an active research area for many years. Several methods have been proposed to automatically identify three broad classes of intent: navigational, informational and transactional [2][11]. Most of these methods rely on additional information such as result clicks or anchor text [9], part of speech tagging [8] or mouse cursor movements on SERPs [6]. For example, Lee et al. [10] used the observation that the click distribution of navigational queries are usually highly skewed toward a few domains to distinguish navigational and informational search queries.

Jansen et al. [7] proposed a rule-based method that solely relied on queries. This supervised method is similar to ours in that it did not utilize additional information beyond queries. However, general rules such as the presence of terms like “download” indicating a transactional query are not applicable in our context since many queries of opposite labels share terms. Broder et al. [4] proposed methods to find relevant advertisements for tail queries. Our two studies were motivated by a similar goal of building effective classifiers for domains where positives are rare. However, our techniques are different. We propose methods to find positives to build more balanced training sets for classification. In contrast, Broder et al. target offline computation on head and torso queries.

While our focus is not on automatic query expansion (AQE), work in that area is still relevant to the research described herein. A survey of the significant body of literature on AQE can be found in [5]. Broder et al. [3] also targeted rare query classification, proposing a pseudo relevance feedback mechanism for classifying a query by



*CIKM'15*, October 19-23, 2015, Melbourne, VIC, Australia
© 2015 ACM. ISBN 978-1-4503-3794-6/15/10…$15.00
DOI: http://dx.doi.org/10.1145/2806416.2806594.

[1] Work performed at Microsoft Research.

first classifying its search results and then using a voting mechanism to determine the label of the query. Their method requires downloading and processing the content of top-ranked results (and of course assumes that these top-ranked results are relevant to the query). Such a method is better suited to query *topic* classification or cases where a document likely satisfies a singular query intent. Topic may help inform query *intent* classification (our task). However, a document (e.g., a page on safe dosing limits for a medicine) can satisfy multiple intents (e.g., a query clearly indicating a user intends to overdose versus checking on how to safely take the medicine), and the query text plays a pivotal role in indicating searcher intentions that is not fulfilled by considering document topic. In contrast to Broder et al. [3], we rely solely on the query text. In these respects, the two methods are quite complementary.

# 3. TRAINING DATA

We now describe how we collected and manually labeled our query classification data. In particular given a small seed set of positive queries, we broaden this set in a two-step process. First, we automatically expand the query set (denoted as *ThreeHop*) using a three-step graph walk on the set of query suggestions obtained from Bing. In the second step, we further expand this to a set of related queries (denoted as *Related*) that are similar to the queries in *ThreeHop* in some way (specifically in applying bigram matching, trigram matching, or session matching).

## 3.1 Expansion through Graph Walk

To follow a common paradigm for constructing query classifiers, we assume that we have a small set of positive trigger queries and seek to broaden these to other relevant queries using commonly available resources. We started with a small seed set of queries that unambiguously expressed self-harm intent and expanded it by including the top-ten related query suggestions returned for those queries by Bing. Query suggestions are a convenient way of expanding an initial query set that is also reproducible publicly through search engine APIs. Focusing on query classification, since query suggestions are often largely influenced by session co-occurrence, this step can be viewed as sampling from the head of session co-occurrence. Our initial seed set was constructed manually and contained the following four queries: [how to commit suicide], [how to kill yourself], [I want to kill myself]**,** and [I want to commit suicide]. Our choice of the seed set queries was quite arbitrary; since we lacked knowledge about the query-space for this domain, we selected four queries that we believed unambiguously express self-harm intent. This may also be quite typical: it is unlikely that search engine designers will have domain knowledge in each domain being targeted for specialized query support. With subsequent expansion, we found several positives which were more specific than the four queries that we started with (e.g., [how to od on xanax], [what is the best combination of pills and carbon monoxide in suicide]). Note that mere synonym substitution can help us find queries such as [how to shoot yourself] from [how to kill yourself], but in our process we also find queries which are related in intent but lack common synonyms with other positives (e.g., [final exit], which is a book on assisted suicide). One iteration of expanding our set involves adding all distinct suggestions for each query. We used a standard publicly available API for such expansion with a user account that had no previous search history (to avoid biasing the query suggestions with personalization signals). We repeat to expand our seed set to 662 queries which includes all neighbors within three hops of the seed set in the query suggestion graph. We wanted our set to be diverse enough while keeping the number of queries unrelated to self-harm intent (e.g., queries related to celebrities who attempted self-harm) restricted. For this goal, we found three hops to be a reasonable heuristic.

**Table 1: Breakdown of the positives obtained.**

| Relatedness Category | Number of positives | *P*(positive) | Pos. Relative to Random |
|---|---|---|---|
| Bigram | 15 | $1.34 \times 10^{-7}$ | 13.4 |
| Trigram | 25 | $1.88 \times 10^{-7}$ | 18.8 |
| Session | 128 | $5.79 \times 10^{-7}$ | 57.9 |
| Random | 1 | $1 \times 10^{-8}$ | 1.0 |

## 3.2 Sampling Related Queries

Given the rarity of positive examples, one challenge in building query classifiers is that if we simply sampled randomly, we would likely find no positives. Conversely, building an evaluation set by only leveraging query suggestions would be overly sensitive to head (popular) queries and our initial seed set. To this end, we expanded the set of queries for labeling in a way that would likely either find positive queries or queries that we would likely erroneously trigger on (false positives) without overly-strong assumptions. We sampled queries in one of three ways. First, we sampled from all search sessions with a co-occurring query in *ThreeHop* (denoted *Session*). This expands the set of queries likely covered by the query suggestions to include more tail sessions. We had access to search logs from Microsoft's Bing search engine and employed the commonly-used practice of demarcating session boundaries via 30 minutes of inactivity. All queries were sampled from the logs of the aforementioned search engine from June 1 – December 31, 2013 in the English-speaking U.S. locale. We also sampled queries from the logs that had at least one bigram or trigram (denoted *Bigram* and *Trigram* respectively) overlapping with the queries in *ThreeHop*. We did this since queries with bigram/trigram overlap with this initial set would be more likely to either be predicted positive/negative.

To further ensure diversity and balance across the set of queries we sought a mechanism to help ensure the related query types were distributed across both query frequency and the likely ambiguity of the query. To achieve this, we sampled 600 queries of each related query type. Within each related query type, we sampled evenly between head-queries (20 or more occurrences in a six-month query log) and tail queries (less than 20 occurrences in a six-month query log). Finally, to help ensure that the set covered a likely range from unambiguously negative to unambiguously positive, we trained an initial classifier (see Section 5) on the labeled *ThreeHop* set, and used that classifier's predictions to further stratify the space. From each decile of the classifier's predicted confidence, we sampled 30 head and 30 tail queries for use in our analysis.

We also sampled 600 queries at random stratified similarly (60 from each decile). Sampling yielded 2,400 queries for the related set (in addition to the 662 queries in *ThreeHop*) After manually labeling these 2400 queries, the *Related* query set consists of 169 positive queries (divided between relatedness categories as shown in Table 2), 2212 negative queries, and 19 queries were labeled as indeterminate. Indeterminate queries were discarded from *Related*.

## 3.3 Labeling

Following the selection of subsets of queries as defined in the previous section, the queries were manually labeled by three labelers. As with any query classification task, deciding how to label a particular query has to be balanced between the specific intent and the likely intent on average across all searchers issuing that query. For example, the query [suicide] is quite general and may express many research intents related to philosophy or ethics courses, medical research, and similar topics. That is, it is not clear that the searcher has a likely intent for self-harm. This is in contrast to the much more specific intentions demonstrated in the seed queries, some of which

**Table 2: Classification performance on *Related*$_{test}$.**

| Measure | Trained on *ThreeHop* | Trained on *ThreeHop* + *Related*$_{train}$ |
|---|---|---|
| Accuracy | 91.06% | 93.29% |
| Precision | 68.75% | 73.33% |
| Recall | 28.94% | 57.89% |
| F1 score | 40.74% | 64.70% |

include first-person language, e.g., [I want to kill myself]. To help provide consistency in judgments, labeling guidelines instructed labelers to look for a likely clear intent of self-harm which includes, among other things, focused questions on suicide methods and their effectiveness, but does not include queries on suicide demographics, celebrity suicide, and various death-related obsessions. Example queries representing each class were also provided to assist the labelers. Labelers were allowed to inspect the search results for a query but instructed to assume that the searcher may not find any results relevant. Labelers reported that examining results was useful to identify special cases of queries such as lyrics or song titles that were not obvious from the query text alone.

Three annotators labeled all of the queries in the *ThreeHop* set. The Fleiss' kappa measure of the inter-rater agreement between the three labelers was 0.73, indicates substantial agreement between the annotators at this task. Disagreements were examined to help clarify the labeling guidelines with additional examples. After resolving disagreements, the *ThreeHop* set comprised 390 positives, 259 negatives and 13 queries labeled as indeterminate. The fact that some labels could not be resolved, even after discussion among the labelers, reflects the difficulty of this task. These uncertain queries were discarded from the *ThreeHop* set during training.

# 4. CLASSIFIER DECISION SURFACE

We now briefly consider the types of classification model and representation that are necessary to accurately model both the self-harm domain and more generally the typical interactions seen in query classification problems. Given our focus on classification using only query text, the primary question that we ask is whether a linear classifier employing a bag-of-words representation or even a bag of n-grams is sufficient? Inspection of the labeled data quickly reveals that there are many interactions which are not likely to be additive. For example, there are subtleties of word ordering. The query [painless suicide] is a likely self-harm query but [suicide is painless] is a song for a very popular television series. This latter query is an example of the types of "exceptions" that often occur where a head query has a very specific intent not obvious from the query text alone. A similar example is [kill yourself] which is also a likely self-harm query versus [kill yourself in 5 minutes] which is a popular online game. Finally, while often a single word or phrase can be pivotal as in [painless easy ways to kill yourself] vs. [painless easy ways to kill mold]. The phrase "kill yourself" occurs in both positive and negative examples as seen in the previous example. Alternatively, "kill mold" covers this negative but misses the more general pattern as this same phrase is seen for many related items, e.g., "….kill bugs", "…kill roaches", etc. Thus, more general pattern matching may help with generalization, but a bag-of-words approach is clearly not sufficient given the complexity of the space. As the representation is extended to increasingly longer n-grams, a classifier can learn the general structure while memorizing exceptions. As a compromise between these extremes we target a representation that employs features of unigrams, bigrams, and trigrams.

# 5. PREDICTING QUERY INTENT

To predict self-harm intent, we first trained an off-the-shelf linear support vector machine (SVM) on the labeled *ThreeHop* set using unigram, bigram, and trigram features. This particular SVM was specifically designed to handle large-scale query classification. This classifier was used to provide the stratified sampling over query class probability for each of the related query methods in Table 1. Since the samples are stratified by posterior, one would expect the number of positives to be similar within each bin across the different methods and the total number of positives to also be the same across the different methods. If the classifier were well-calibrated we would expect the number of positives to match the expected posterior, but even if it did not, a consistent miscalibration would skew each related query method in the same way. If they skew differently, it suggests that building a query classifier by simply sampling suggested queries is subject to bias that can be partially overcome by broadening sampling as described herein.

Table 1 (on the previous page) shows that we obtained far more positives using *Bigram*, *Trigram* and *Session* approaches than via the plain random sampling approach. By using the frequency of bins for stratification, we can estimate the overall probability of positive according to each method. We use the random positive rate as the expected rate of positives for the proportion not matching a related filter. Interestingly, the variance across these estimates demonstrates the difficulty in sampling for rare items. The final column expresses these as a ratio to random to emphasize the magnitude of variation and the greater rate of positive discovery. This result underscores that by plain random sampling we would get very few positives, meaning that the classifier would have limited data for further improvement. Among the three types of relatedness we considered, *Session* obtained the maximum number of positives which follows the intuition that a person with a specific intent will make similar searches within a given session.

We split *Related* for train and test (90:10, *Related*$_{train}$ and *Related*$_{test}$) and trained the SVM on *Related*$_{train}$ + *ThreeHop*). As a baseline, we compared the performance with a classifier only trained on *ThreeHop*. Table 2 shows that with additional training data, we obtained improvement over all measures of classification performance. Since we have widened the net of different types of self-harm queries through *Related*, our recall performance substantially improves. A classifier that always predicts negative (the marginal) only achieved an accuracy of 89.38%. While our queries are sampled in a manner to help discover positives and likely errors, we should also consider operative performance. Such performance is weighted by query frequency and the most frequent queries are often easily classified. Even in our challenging set, we see better performance when reweighting by query frequency with precision, recall, and F1 of 92.64%, 83.31%, and 91.83% respectively.

On analyzing the misclassified queries, we found that some of them happened as a result of possible typos (e.g., [hbest ways to commit suicide], [how to commit suciside]) and can be accurately classified if the typo is corrected (e.g., by applying automated spelling correction). Some misclassifications can be handled with appropriate domain knowledge. For example, a typical pattern for self-harm queries was found to contain medicines that are used for overdosing (e.g., [lorazepam suicide]). Adding such lists of medicines in the domain knowledge of the classifier and treating them as a wild-card can reduce the number of misclassifications. Some of the misclassified queries were long queries (e.g., [what is the best combination of pills and carbon monoxide in suicide]). Adding higher order n-grams may address these type of queries.

**Table 3: Sampling techniques for active learning.**

| Sampling Technique | Strategy |
|---|---|
| Batch random | Pick *k* samples at random |
| Batch one-sided extreme | Pick *k* samples with predicted class probability closest to 1, breaking ties arbitrarily |
| Batch two-sided extreme | Pick *k* samples with predicted class probability closest to 0 or 1, breaking ties arbitrarily |
| Batch maximum uncertainty | Pick *k* samples with predicted class probability closest to 0.5, breaking ties arbitrarily |

In various learning scenarios, active learning is found to be useful in reducing the number of training examples required to learn a concept and thus being particularly useful when labeling resources are scarce. Also, in a practical setting, requesting labels in batches is more cost-effective than requesting one at a time. For these reasons, next we analyzed the performance of different sampling strategies for active learning in a batch setting. In doing so, we were interested in whether traditional active learning approaches can succeed in this setting or whether it is primarily the discovery of extreme positive/negative examples that drives classifier improvements. We considered strategies listed in Table 3. We chose random sampling as our baseline and maximum uncertainty since it is a known sampling technique that is useful across many different active learning tasks. We observed that several queries classified as positives with high confidence were actually a negative query with a high degree of n-gram matching with self-harm queries (e.g., [painless easy ways to kill mold]). In order to examine if labeling queries predicted with very high or low confidence can improve performance, we explored two more strategies: one-sided extreme and two-sided extreme. We choose one-sided extreme since it was found to outperform maximum uncertainty sampling in the context of short document classification [1][12]. Figure 1 plots the area under the receiver operating characteristic (AUROC) curve comparison of different sampling strategies. We initially train our SVM on *ThreeHop* and at each step, we add 10 samples from $Related_{train}$. We restrict ourselves to 350 additional samples since the earlier part of the curve is mainly interesting as the performance of each strategy will eventually converge. Since ties are broken randomly, which can create some amount of variability, we repeated the experiment five times for each sampling strategy and report the average performance. Figure 1 shows that maximum uncertainty outperforms other the techniques (in many cases with statistical significance at $p < 0.05$ using *t*-tests), and there is no clear winner among batch two-sided extreme and random sampling. So we conclude that active learning does help in improving performance, but only identifying one-sided or two-sided extremes do not outperform maximum uncertainty sampling.

## 6. DISCUSSION AND CONCLUSIONS

We proposed a structured approach to building a classifier when positives are rare and we have a small initial set of positive examples. We showed that our approach finds many more positives than random sampling. We focused on self-harm, an important domain underexplored in IR despite the use of search systems by people interested in learning about self-harm, and we share valuable insights about our key challenges. We compared the performance in a batch active learning setting and found that maximum uncertainty is the best sampling strategy given limited labeling resources.

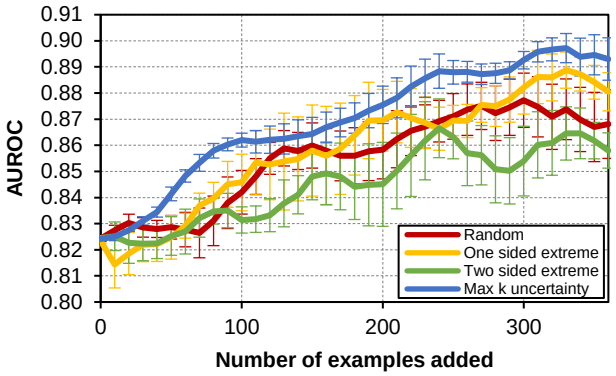


**Figure 1: Performance of different query sampling strategies. Lines denote five-point moving average over five runs (±SEM).**

Although we focused on self-harm, most of the challenges faced in our work apply to any query classification task with rare positives. We do not address the broader question of when given self-harm queries or the high probability of a self-harm intent, search engines should show an answer or offer other interventions. We hope this research prompts a broader discussion of self-harm intentions within relevant communities, including IR, mental health, and ethics, about if/when/how such interventions should be triggered.

## 7. REFERENCES


[1] Attenberg, J., Melville, P., and Provost, F. (2010). A unified approach to active dual supervision for labeling features and examples. In *Machine Learning and Knowledge Discovery in Databases*, 40–55.

[2] Broder, A. (2002). A taxonomy of web search. *SIGIR Forum*, 36(2): 3–10.

[3] Broder, A., Fontoura, M., Gabrilovich, E., Joshi, A., Josifovski, V., and Zhang, T. (2007). Robust classification of rare queries using Web knowledge. *SIGIR*, 231–238.

[4] Broder, A., Ciccolo, P., Gabrilovich, E., Josifovski, V., Metzler, D., Riedel, L., and Yuan, J. (2009). Online expansion of rare queries for sponsored search. *WWW*, 511–520.

[5] Carpineto. C. and Romano, G. (2012). A survey of automatic query expansion in information retrieval. *CSUR*, 44(1): 1.

[6] Guo, Q. and Agichtein, E. (2008). Exploring mouse movements for inferring query intent. *SIGIR*, 707–708.

[7] Jansen, B.J., Booth, D.L., and Spink, A. (2008). Determining the informational, navigational, and transactional intent of Web queries. *IP&M*, 44(3): 1251–1266.

[8] Kang, I.H. and Kim, G.C. (2003). Query type classification for web document retrieval. *SIGIR*, 64–71.

[9] Li, Xiao, Wang, Y.-Y., and Acero, A. (2008). Learning query intent from regularized click graphs. *Proc. SIGIR*, 339–346.

[10] Lee, U., Liu, Z., and Cho, J. (2005). Automatic identification of user goals in web search. *WWW*, 391–400.

[11] Rose, D.E. and Levinson, D. (2004). Understanding user goals in web search. *WWW*, 13–19.

[12] Sindhwani, V., Melville, P., and Lawrence, R.D. (2009). Uncertainty sampling and transductive experimental design for active dual supervision. *ICML*, 953–960.