2017

A comparison of automatic search query enhancement algorithms that utilise Wikipedia as a source of a priori knowledge

Kyle Goslin

Markus Hofmann

A Comparison of Automatic Search Query Enhancement Algorithms That Utilise Wikipedia as a Source of A Priori Knowledge

Kyle Goslin*
Institute of Technology Blanchardstown
Dublin 15, Ireland
kylegoslin@gmail.com

Markus Hofmann
Institute of Technology Blanchardstown
Dublin 15, Ireland
markus.hofmann@itb.ie

ABSTRACT

This paper describes the benchmarking and analysis of five Automatic Search Query Enhancement (ASQE) algorithms that utilise Wikipedia as the sole source for a priori knowledge. The contributions of this paper include: 1) A comprehensive review into current ASQE algorithms that utilise Wikipedia as the sole source for a priori knowledge; 2) benchmarking of five existing ASQE algorithms using the TREC-9 Web Topics on the ClueWeb12 data set and 3) analysis of the results from the benchmarking process to identify the strengths and weaknesses each algorithm.

During the benchmarking process, 2,500 relevance assessments were performed. Results of these tests are analysed using the Average Precision @10 per query and Mean Average Precision @10 per algorithm.

From this analysis we show that the scope of a priori knowledge utilised during enhancement and the available term weighting methods available from Wikipedia can further aid the ASQE process. Although approaches taken by the algorithms are still relevant, an over dependence on weighting schemes and data sources used can easily impact results of an ASQE algorithm.

CCS CONCEPTS

• **Information systems → Information Retrieval**;

KEYWORDS

Search Query Enhancement, Information Retrieval, Wikipedia

ACM Reference Format:
Kyle Goslin and Markus Hofmann. 2017. A Comparison of Automatic Search Query Enhancement Algorithms That Utilise Wikipedia as a Source of A Priori Knowledge. In *FIRE'17: Forum for Information Retrieval Evaluation, December 8–10, 2017, Bangalore, India.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3158354.3158356

*corresponding author.

FIRE'17, December 8–10, 2017, Bangalore, India

ACM ISBN 978-1-4503-6382-2/17/12…$15.00
https://doi.org/10.1145/3158354.3158356

1 INTRODUCTION

The art of search is a personal experience. With this, search results that may suit one user, may not be the results another user is expecting. During search, the abilities of users can also be brought into consideration, as the experience level can differ between users [11]. The length and quality of queries entered by users can also be questionable as often queries are too short and others are too verbose [3], leading to difficulty during the Information Retrieval (IR) process. Automatic Search Query Enhancement (ASQE) is the process of enhancing a user search query, regardless of length or content, in an effort to return more meaningful results to the user [17]. ASQE algorithms often depend on a source of a priori knowledge to aid the term selection and weighting process. Recent ASQE algorithms [2, 4, 5, 20, 23] have shown the use of dynamic data sources, such as Wikipedia, that offer high quality and ever changing articles with common fields and structure, can be beneficial to the ASQE process.

Unlike other data sets, Wikipedia offers a selection of tools that can be harnessed to aid the term weighting and selection process. These include the Wikipedia search API[1] and article backlink API.[2] Each Wikipedia article is delivered in a defined HTML structured format, with common headings and sections, further aiding the extraction of relevant data.

This paper describes the benchmarking and analyse of five current ASQE algorithms that utilise Wikipedia as the sole source for a priori knowledge. Each of these algorithms is focused upon the utilisation of available Wikipedia data set and APIs. An overview of the five selected algorithms that utilise Wikipedia to aid the ASQE process are described in Section 2. For each algorithm, the core functions are described along with the key elements of Wikipedia used. To ensure consistency and repeatability of results, Section 3 describes the methodology that was followed during this research. The data sets used for retrieval are described along with the relevance assessment methods used.

The results of the testing and analysis performed can be seen in Section 4. During this analysis, an additional focus is placed on both, long and short queries, as typically they are most difficult to gauge the user's intent. A sample of the enhancement terms generated by each algorithm are then shown. Section 5 discusses the results of each algorithm, providing a detailed look at the success and failings of each. This paper concludes in Section 6 with a final summary of the findings.

[1] https://www.mediawiki.org/wiki/API:Search
[2] https://www.mediawiki.org/wiki/API:Backlinks

2 RELATED WORK

ASQE is the process of automatically enhancing a user search query, typically through the addition, removal or correction [17] of search terms to improve precision / recall of a search query. Ongoing research [23] has continued to show that Wikipedia is useful as a source of prior knowledge to aid ASQE algorithms due to the quantity and wide domain of topics available. Unlike utilising static document collections and thesauri[3] that require expert knowledge to maintain, Wikipedia has shown to be beneficial as a source of prior knowledge for domain specific query enhancement such as in the area of patent retrieval [1, 16].

As ASQE is built upon a number of different IR techniques, each component of the ASQE process can be further enhanced by utilising Wikipedia; such as term weighting [10], linguistic understanding [15], relevance calculation [22], term disambiguation [7, 19] and similarity assessment based up Wikipedia articles [9]. The additional data available in Wikipedia can be beneficial to users, as during search, users with little knowledge about the area of search have shown to perform worse due to their lack of prior knowledge when compared to domain experts [12]. He and Ounis [8] identified two possible reasons for the failure of ASQE, low query quality and topic drift. As search queries can be overly simple or complex, recent research has moved towards understanding the structural and syntactic complexity of search queries [14], which can further improve ASQE techniques. The context of a user during search plays an important role as it often does not exist for the user at the beginning of a search session [6].

ALMasri et al. [2] proposed a Wikipedia based semantic query enrichment algorithm, whereby semantically related terms are extracted from Wikipedia and then used as Pseudo Relevance Feedback (PRF). This process is achieved through the following steps: Collect all articles $S(q)$ which are entitled by the user's query q. Each article $a \in S(q)$ has the probability $P(a \mid q)$ of being used in the enrichment process. The probability is defined as $P(a \mid t) = \frac{|O(a)|}{\sum_{a_i \in S(t)} |O(a_i)|}$, where $O(a)$ is the set of articles that a points to. The expansion set ES of selected n number of articles for user query q are defined as $ES(q, n) = \bigcup_{a \in S(q)} f(a, \lceil n \times P(a \mid q) \rceil)$. The collection of terms for query q are built from a union of article titles in the enrichment set. A weight is attached to each between 0 and 1, whereby 1 is most important and 0 is least important. The weight for each of the terms is defined as $weight(t, q_e) = \alpha \times SIM(a_q, a_t)$, whereby α is a tuning parameter between 0 and 1. The similarity calculation between two articles, a_1 and a_2, is defined in Equation 1, where $I(a)$ is the set of articles that points to a.

$$SIM(a_1, a_2) = \frac{\mid I(a_1) \cap I(a_2) \mid + \mid O(a_1) \cap O(a_2) \mid}{\mid I(a_1) \cup O(a_1) \mid + \mid I(a2) \cup O(a2) \mid} \quad (1)$$

Boston et al. [4] proposed a tool titled Wikimantic which exploits Wikipedia articles and their inter-article reference relations which has shown to be effective for short queries. They define an AtomicConcept as a simple form of a concept. Each article is considered a series of terms which was generated by an AtomicConcept. The prior probability of $P(A)$ generating terms, where A is an individual article is defined as $P(A) = \frac{number\ of\ incoming\ links}{number\ of\ links\ in\ Wikipedia}$. As most of the articles in Wikipedia are linking to other articles, the authors define the probability of article A generating term t is defined as $P(t \mid A) = \frac{count(t,A)}{number\ of\ words\ in\ A}$.

Due to the limitation that not all articles will have a variety of different terms to check their probability with, the Microsoft n-gram corpus[4] containing 100,000 unique terms is used. Building upon an AtomicConcept, a new variant is defined as a MixtureConcept which is a collection of different AtomicConcepts. A MixtureConcept is defined as $M = \{(w_i, A_i) \mid i = 1...n\}$, where w_i is the weight of the concept and A is an individual article concept. In this Equation, i is the current AtomicConcept being viewed inside of M and w_i is the weight of A_i in MixtureConcept M. The probability of a MixtureConcept, $P(M)$ generating terms is defined as $P(M) = \sum_{i=1}^{n} w_i * P(A_i)$. The probability of generating term t for mixture concept M is defined as $P(t \mid M) = \sum_{i=1}^{n} w_i * P(t \mid A_i)$. After an AtomicConcept set has been generated, a weight w_i is applied. S is the number of terms in the Concept. This is shown in Equation 2 and the probability of $P(A)$ generating term t is shown in Equation 3.

$$w_i = P(A_i \mid S) = \prod_{j=1}^{|S|} P(A_i \mid t_j) \quad (2)$$

$$P(A_i \mid t_j) = \frac{P(t_j \mid A_i) * P(A_i)}{P(t_j)} \quad (3)$$

Given query Q, a set of MixtureConcepts are created and then Equation 4 is used for generating possible expansion terms where $P(t \mid A_i)$ is the likelihood of generating term t from the AtomicConcept A_i and w_i is the weight of the AtomicConcept A_i in MixturenConcept M for user submitted query Q described as $M(Q)$. N is the number of documents in the collection and $df(t)$ is the number of documents that contain t. $ln\frac{N+1}{df(t)}$ is the described as the IDF weighting for the given term t.

$$ExpWeight(t \mid M(Q)) = \sum_{A_i \in M(Q)} P(t \mid A_i) \times w_i \times ln\frac{N+1}{df(t)} \quad (4)$$

Xu et al. [18] outlined a query dependant PRF approach based on Wikipedia. They first began with an approach to categorise user queries into three categories, entity queries, ambiguous queries, and broader queries. They also proposed a number of different approaches for enhancement including a Relevance Model based approach, Field Evidence approach utilising the fields identified in a Wikipedia page and an entity page based approach. Their results show that the entity page based approach was the most successful and is discussed below. Instead of focusing on the top ranked documents as shown above, this approach focuses on utilising the entity page, e.g., the page which corresponds directly to the topic that the user is searching as an initial source of additional terms for PRF. The procedure followed during this study is defined as: 1) Identify an entity page for the user submitted query Q, 2) All terms on the entity page are ranked using TF-IDF, and 3) Top k terms are extracted, where the score for a term t on an entity page is defined using TF-IDF, where tf is the TF on an entity page and idf is computed as $log(N/DF)$, and n is the number of documents

[3] https://wordnet.princeton.edu/

[4] http://research.microsoft.com/apps/pubs/default.aspx?id=130762

in the Wikipedia collection and DF is the number of documents that contain term t defined as $score(t) = TF - IDF$.

Bruce et al. [5] described query expansion powered by Wikipedia hyperlinks. This approach begins by first breaking a query into query aspects. Poorly represented areas of the query are then enhanced with additional terms. This process contains six steps outlined as: 1) The user's initial query is received, 2) aspects of the query are identified, 3) Wikipedia articles are selected, 4) aspect vocabulary is constructed, 5) finding under represented aspects, and 6) query expansion. For aspect identification, Link Probability Weighting is used. This is done by counting the number of documents where the term is already a hyperlink divided by the number of documents where the term appeared. Aspects are selected from the highest value through to the lowest. An aspect is ignored if it is a subset of an already selected aspect. No aspects with weighting of 0 should be added, unless they contain terms that are yet to be covered by selected aspects. Aspect Identification is complete when each term of the query has been covered by an aspect. A collection of articles is created with a connection to the aspects defined. Each of the aspects are disambiguated individually using Link Probability measure. A cut-off threshold is then utilised, and all articles that have a confidence greater than half of the maximum measure are added. Aspects are disambiguated into pairs using the Wikipedia Link Based measure, described in Equation 5. The similarity between two articles is defined, where A and B are the set of articles that link to a and b and W is the entire Wikipedia collection.

$$sr(a,b) = \frac{log(max(|\ A\ |, |\ B\ |)) - log(|\ A \cap B\ |)}{log(|\ W\ |) - log(min(|\ A\ |, |\ B\ |))} \tag{5}$$

Aspect vocabulary construction aims to build a weighted vocabulary for each aspect using the article set created previously. All terms appearing in the selected articles are vocabulary candidates weighted by their relation to their corresponding aspects. Each candidate term is added along with the Wikipedia Link Based Measure score. Finally, under represented aspects are identified. This step selects the best expansion term by counting term frequencies of all terms in the first 10 documents of an initial Bing[5] search. The first 50 highest weighted terms are normalized. Scores are then calculated by multiplying term weights in the aspect vocabulary by their frequency weighting in the query vocabulary. From this the lowest score is determined to be the least represented. The aspects vocabulary is assigned as the final output for query expansion.

Zhao et al. [20] described a novel term semantic query model based on Wikipedia. This approach is focused upon finding the semantic relatedness between terms using Wikipedia. The semantic correlation of to terms is defined as $T_jW_i = TF_i * log(\frac{N_1+N2}{n})$, where i,j = 1,2 and T_jW_i represents the weight of the ith common words in T_j word groups. The summary paragraph available for all Wikipedia articles is used to compute the semantic relatedness of two terms shown in Equation 6, where a,b represents two terms that are used for semantic computing and T_i, T_2 represent the word group obtained by word segmentation on the summary paragraph, N_1, N_2 are the number of words in word group T_1, T_2.

$$sim_a(a,b) = \frac{MAX(N_1,N_2)}{MIN(N_1,N_2)} + \sum_{i=1}^{n} T_i w_i * T_2 W_i \tag{6}$$

[5]https://www.bing.com/

Semantic link relatedness is computed using Equation 7, whereby a and b represent two terms that are used for semantic computing, A is the number of inbound links for term a and B is the number of inbound links for term b. W is defined as the number of individual articles in Wikipedia.

$$sim(a,b)_{in} = \frac{log(MAX(|\ A\ |, |\ B\ |)) - log(|\ A \cap B\ |)}{log(|\ W\ |) - log(\min(|\ A\ |, |\ B\ |))} \tag{7}$$

Zhao et al. [21] described a method for Named Entity Disambiguation, which contains a query expansion based upon the utilisation of Wikipedia terms based on co-occurrence mentions. The authors describe that often, in the case of an article, a name is mentioned in complete form at the start of an article. Two main strategies for identifying candidates are: 1) queries that contain abbreviations, a match is made to terms which have similar capitalisation; 2) queries that contain continuous strings where the first letter of the string is also a capital letter, a match can be made to a candidate. The Wikipedia data utilised by this approach includes article titles, article content and article redirections. In their method, an initial query is placed to collect the top-k documents for a given query. Any candidate terms that are identified in the article collection become part of the collection of enhancement terms and articles returned become part of the article collection. The authors of this method identify that their query expansion approach is simplistic, and titled it the feedback-query-expansion method, as it incorporates a feedback loop to find candidates during retrieval. Due to the simplistic nature of this approach, it will be excluded from the testing described in this paper.

Zingla et al. [23] described the issue of short queries in microblog retrieval and implemented ASQE using Wikipedia. The authors' method of identify candidate expansion terms was done by, 1) selecting unstructured full texts related to the original query utilising TF-IDF to select similar texts, 2) texts are tagged using the TreeTagger, 3) extract nouns from text, and 4) generate association rules using the CHARM algorithm. After a collection of candidate terms has been generated, additional processing is performed to ensure the terms are related to the original query terms. This is done through the use of their proposed semantic relatedness measure titled ESAC which combines Explicit Semantic Analysis using Wikipedia and association rules' confidence measures. This is described in Equation 8 where $Conf_{max}(R,q,w)$ is the max of the confidence of any association rule from R_j from rule collection R. $ESA(q,w)$ is the score of relatedness between the query q and the candidate term w and α is a tuning parameter between 0 and 1.

$$ESAC(q,w) = \begin{cases} (\alpha \times ESA(q,w) + (1-\alpha) \times Conf_{max}(R,q,w) \\ if Conf_{max}(R,q,W) \neq 0; \\ ESA(q,w), otherwise. \end{cases} \tag{8}$$

After this process is completed, the most related terms are then added to the original search query. Their research showed that the best results were achieved with rule mining and a term filtering phase which used a Wikipedia-based ESAC to prevent similar terms being utilised in the enhanced query.

3 METHODOLOGY

Each of the selected algorithms were provided 50 of the TREC-9 Web Topics[6] for enhancement. After the enhancement was performed, the resulting enhanced query was passed to the ClueWeb12 full data set[7] Batch Query Service[8] to retrieve documents for relevance assessment. In this research, the testing and analysis methodology followed during the enhancement and ranking of search topics for each algorithm are defined as:

(1) Select test topic, Q, from test query collection.
(2) Pass Q to the current enhancement algorithm under analysis.
(3) Gather 10 generated terms from the selected enhancement algorithm.
(4) Merge original query *Q* and the new additional enhancement terms.
(5) Pass the enhanced query to the ClueWeb12 full data set Batch Query Service to retrieve results.
(6) Calculate the Average Precision @10 for the given enhanced query based on the results returned.

The Average Precision @10 was calculated by first analysing the top ten results returned per enhanced query from the ClueWeb12 full data set Batch Query Service running the Lemur IR engine[9]. The topic description was then gleaned from the description field for each topic from the given TREC-9 Web Topic collections If the result returned was relevant, the result was marked with 1, if the result was irrelevant it was marked as 0. For each test completed, 500 manual relevance assessments were performed. In addition to the Average Precision @10, the Mean Average Precision @ 10 for each test was also calculated providing an overall score for each algorithm tested. Existing relevance assessments for the TREC-9 topic collection were not used as the larger ClueWeb12 data set was used. In addition to this, many existing algorithms described in this research used alternative data sets and test topics, providing difficulty during comparison to existing author results.

As research in the area of ASQE algorithms based on Wikipedia is limited, out of the six algorithms outlined in Section 2, the Algorithm by Zingla et al. [23] was omitted as it was based on Rule Mining, which was conceptually distant from the algorithm proposed in this paper. The five remaining algorithms were chosen for analysis. For each enhancement algorithm analysed, 10 enhancement terms were added. This was based upon research by Ogilvie et al. [13] which outlined that 10 or less provided the optimal enhancement. As this research is not focused on optimisation of this parameter, 10 enhancement terms was chosen for each tested algorithm. Table 1 describes the algorithms tested, authors and Wikipedia data set components utilised by each algorithm.

4 RESULTS

For each algorithm, the Mean Average Precision (MAP) was calculated using the Average Precision (AP) scores for each of the 50 TREC-9 Web Topics on the ClueWeb12 batch query service. Table 2 provides an outline of these results. The overall standard deviation for each algorithm was calculated on the AP, shown as *STD*. To provide an understanding of each algorithm for short and long queries the STD, and MAP was calculated for 1 term topics (short), 2 term topics (short) and greater than 2 topics (long).

[6] http://trec.nist.gov/data/topics_eng/topics.451-500.gz
[7] http://www.lemurproject.org/clueweb12.php/
[8] http://boston.lti.cs.cmu.edu/Services/clueweb12_batch/
[9] https://www.lemurproject.org/

Table 1: Tested Algorithms

	Authors	Wikipedia Components
Alg 1	Almasri et al.	Search API, Article Titles and Article Content
Alg 2	Boston et al.	Inter-article Link References and Article Content
Alg 3	Xu et al.	Article Content, Search API and Wikipedia Document Collection Size
Alg 4	Bruce et al.	Inter-article link references and Article Content
Alg 5	Zhao et al.	Search API, and Article Summary Text

Table 2: MAP @10 Results from Tested Algorithms

	MAP @10	STD	1 Term MAP	1 Term STD	2 Term MAP	2 Term STD	> 2 Term MAP	> 2 Term STD
Alg 1	0.634	0.415	0.739	0.340	0.536	0.463	0.617	0.438
Alg 2	0.384	0.404	0.636	0.426	0.323	0.337	0.282	0.378
Alg 3	0.714	0.395	0.786	0.356	0.914	0.192	0.549	0.450
Alg 4	0.469	0.462	0.595	0.434	0.442	0.447	0.382	0.479
Alg 5	0.480	0.436	0.671	0.430	0.243	0.413	0.535	0.410

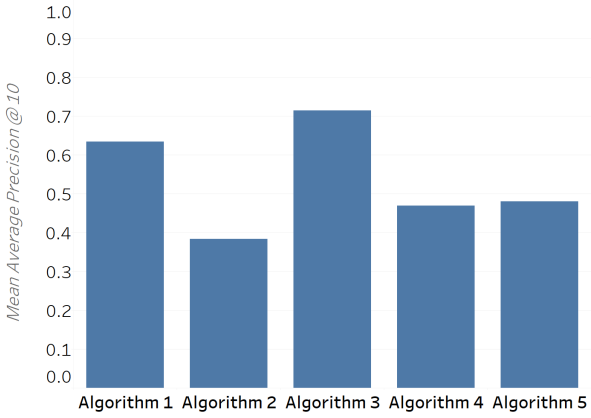

Figure 1: MAP @10 Scores For All Algorithms

Figure 1 shows the MAP scores achieved by each algorithm. In these results we can see that algorithm 3 achieved the highest MAP score. The success of this algorithm can be placed upon the utilisation of entity pages, each representing a single Wikipedia article. This narrowed the scope of terms that the algorithm was working with. This is followed by algorithm 1, with a MAP score of 0.635. The worst performing algorithm overall, was algorithm 2,

with a MAP score of 0.384. Although some of the terms generated by this algorithm were relevant, these were hurt by other terms as a large proportion of the terms were conceptually distant leading to a query drift.

Table 3 provides an outline of the MAP scores achieved by each algorithm and the elements of Wikipedia that were used.

Table 3: MAP @10 Scores for Each Tested Algorithm and Wikipedia Components Used

	MAP @10	Search API	Article Titles	Article Content	Inbound Links	Outbound Links	Summary Text	Article Redirection
WNSSA	0.800	✓	✓	✓	✓	✓	✓	✓
Algo 1	0.634	✓	✓	✓	✓	✓	-	-
Algo 2	0.384	-	✓	✓	✓	-	-	-
Algo 3	0.714	✓	-	✓	-	-	-	-
Algo 4	0.469	✓	-	✓	✓	-	-	-
Algo 5	0.480	✓	-	✓	✓	✓	✓	-

4.1 N-Term Analysis

In this section, a selection of the short queries analysed are described, as short queries force the expansion algorithm to guess context and intent of the query. The short number of terms prevents a large skew from occurring during enhancement. However, as seen in previous sections, due to the lack of context they can also be seen problematic forcing the algorithm to select the context for the user. Figure 2 outlines the AP @ 10 results for all five algorithms with a focus on the single query terms. The best performing short queries included Query 15: *deer*, Query 20: *mistletoe*, Query 39: *calcium*. The reason for the success of these queries can be placed on the fact that each term mainly has a single meaning.

The worst performing single term queries included Query 3: *hunger*, which was focused on the film and not the human state of hunger and Query 44: *nirvana*. In the case of nirvana, this can either be the rock band nirvana or the Buddhist state of enlightenment. Again the direction the algorithm took during the enhancement process was the sole factor in failing the enhancement process.

Figure 3 describes the AP @ 10 results of two term queries. The best performing queries are Query 4: *Parkinson's disease*, Query 48: *hair transplant*, Query 49: *pool cue* and Query 50 *DNA Testing*.

The worst performing queries are Query 7: *Chevrolet trucks* and Query 11: *lava lamps*. The nature of the term *lava* caused many of the algorithms to return results about volcanic eruptions. Query 19: *Steinbach nutcracker* had issues with the name *Steinbach*. This name has a number of different representations depending on the context. These include the location *Steinbach, Manitoba* and also the popular piano maker *Steinbach*. Query 41: *Japanese Wave*, caused many issues as when the two terms are interpreted independently a focus was placed on *Japanese* and also on *Wave*, that when run separately, skew the results. The difficulties with these queries can often be seen with the segmentation of the two terms in the query removing the original context between the terms during enhancement.

4.2 Sample Enhancements

Table 4 provides an overview of the generated enhancement terms for each of the tested algorithms for TREC-9 Topics 4, 17, 31 and 32. In this table we can see that for Topic 4 *parkinsons disease*, algorithm 2 had only a single related term, *neurology*. Algorithm 3 included the initials *pd*, and different elements related to the disease. Algorithm 5 however included terms such as *tuberculosis* and *pathology*, that are not directly related to the original topic.

For Topic 17: *dachshund dachshunds "wiener dog"*, algorithm 1 produced irrelevant results and algorithm 2 produced poor results not related to the topic. Algorithm 3, focused on different breeds that are available and algorithm 5 added in two terms which were irrelevant, *comedy* and *deer*.

Topic 31: *What did Babe Ruth do in the 1920's*, has very narrow room for error as it is specifically looking for one topic, Baseball. Algorithm 2, included terms such as *Curse of the Bambino*, which is related to Babe Ruth, and the name of baseball teams. However, no explicit reference to baseball was included. Algorithm 2, performed poorly overall. Algorithm 3 focused again on the term *Baseball* and outlined other teams and notable names in Baseball. Algorithm 4 did not produce any useful enhancements. Algorithm 5 again outlined *baseball, pitcher* and outlined 1920s related topics such as *prohibition*.

For Topic 32, *where can i find the growth rate for the pine tree?*, algorithm 1 produced terms such as *Christmas tree*, that can be deemed irrelevant, hurting precision. Algorithm 2 produced, terms such as *nigra* and *petals* that can impact the precision. Algorithm 3 produced terms that are relevant to trees such as *pinus* and *cones*, algorithm 5 focused on the genus of trees, although relevant, can hurt the intent of the query.

5 DISCUSSION

Algorithm 1 by ALMasri et al. [2] focused on the utilisation of Wikipedia article titles as a source of expansion terms. A common theme that was often seen in these results is the overwhelming number of function words that are included which may cause a skew in the results. Another issue which is apparent is that titles of articles may be included if they contain one of the terms which the user has added. An example of this is the title of the popular HTTP Web server Tomcat appearing in results. As the term *Cat* appears in the query *Q1 - What is a bengals cat?*. No discrimination is added validating the domain of the article that has been added to the set, it is purely based on the article title. This can be seen as one of the main flaws in this algorithm impacting the results.

Algorithm 2 described by [4] focused on applying weights to individual concepts. Although the terms are relevant to the domain, and more precision was added into the process of selecting the importance of terms, no real understanding is gained about the query that has been entered by the user. A high dependency is placed on the initial retrieval to generate a relevant collection of document which can then be used as a strong base for candidate terms. The coverage of the terms can be seen as very broad, as the

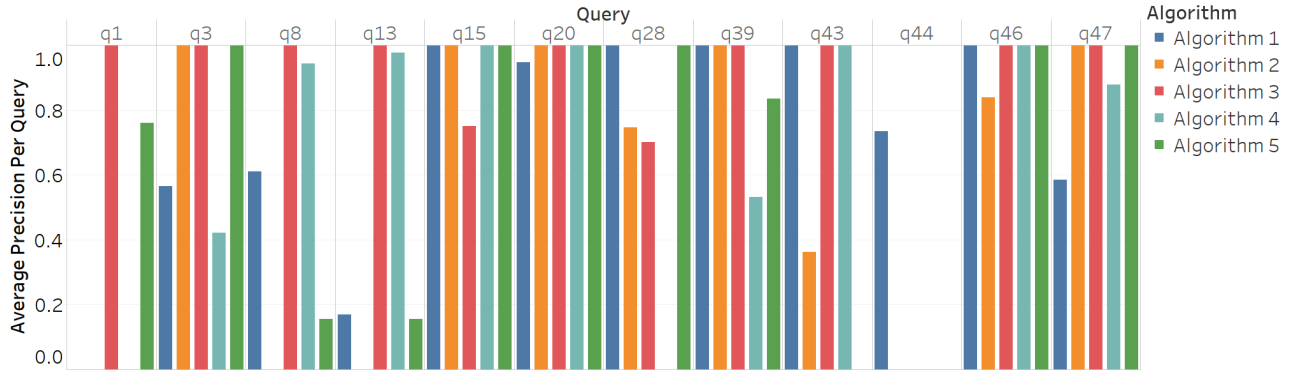

Figure 2: 1 Term Average Precision @10 Scores

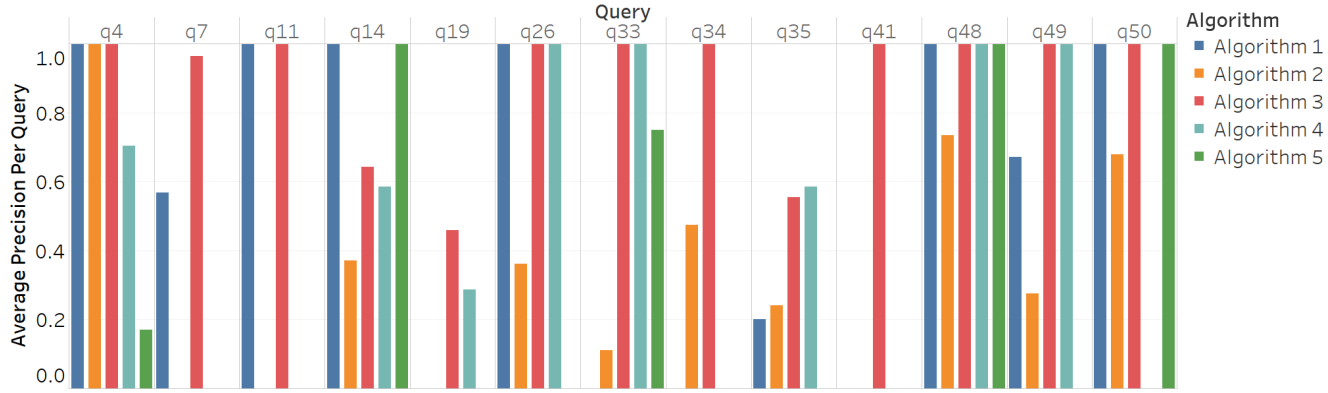

Figure 3: 2 Term Average Precision @10 Scores

number of documents that are included may cover many different domains.

Algorithm 3, described by Xu et al. [18] produced the best results. The overall success of Algorithm 3 can be placed on the simplicity of the algorithm. Rather than utilising a traditional TF-IDF approach that uses a document subset during the calculations, the entire of the Wikipedia collection is used. A useful element of this approach is the focus placed on using the entity page. This focus prevents completely irrelevant terms from making their way into the final enhancements.

Algorithm 4, described in a paper by Bruce et al. [5], was overall the second worst performing algorithm. A number of queries performed badly overall, with barely relevant results. An initial perceived advantage of this approach is the variety of terms that are added. Each of the terms, due to the fact they come directly from Wikipedia pages, has very high quality. A downside to this approach, however, is although the terms are relevant in some contexts, they may not be directly relevant to the query at hand. The intent of the query the user has entered has no impact on the enhancement process.

Algorithm 5, described by [20], focused on the utilisation of inter-wiki links. A heavy dependence is placed on these links. If irrelevant articles are found but have a high number of similar outbound links, then these terms will be given a high weighting which can impact the results. Unlike other algorithms, there is a higher distribution of good and bad results for each of the queries. Many of the terms added were of high relevance in a very broad sense.

From this analysis, the main issues that are impacting the results of algorithms include:

- Naïve addition of documents as sources of knowledge without proper understanding of the domain of the documents that are being added.
- No validation of terms as being relevant or irrelevant as a reputable source of content.
- Content in documents which are not relative to the article, e.g., advertisements, long additional text descriptions.
- Inclusion of function words: Many of the algorithms allow function words or pages which are relevant to Wikipedia to appear in enhancement terms.
- An over dependence on the weights that have been assigned to the terms without additional processing. Although some terms may be very high quality, a post processing stage would greatly improve the overall success of the enhancements.

Table 4: Sample Generated Enhancement Terms for Each Tested Algorithm using TREC-9 Topics

	Topic 4: parkinson's disease
Alg 1	Lewy body disease Parkinson disease American Parkinson Disease Association
Alg 2	home high group including university samii neurology increase list association
Alg 3	pd symptoms levodopa pmid doi dopamine motor disorder lewy brain
Alg 4	dopamine parkinsonism cases therapy sleep studies medication disord although system
Alg 5	neurology parkinsonism psychiatric idiopathic symptom pathology infectious pain tuberculosis pathogenic
	Topic 17: dachshund dachshunds "wiener dog"
Alg 1	Fatal dog attacks in the United States Capitalist pig-dog
Alg 2	result recognized making entering ramirez chase companion essays heed quirk
Alg 3	dapple kennel breed wire-haired miniature akc teckel piebald standard anglo-fran
Alg 4	wire-haired anglo-franxc long-haired smooth-haired california short-haired full-size long-bodied double-dapple merriam-webster
Alg 5	kennel breed comedy deer scent
	Topic 31: what did babe ruth do in the 1920's?
Alg 1	Harmonica Incident Curse of the Bambino Charlie Gehringer The Yankees
Alg 2	yugoslav patrol soap bob miguel arkansas pioneer sabina pop prime
Alg 3	yankees creamer montville baseball sox wagenheim home runs reisler gehrig
Alg 4	economic publishes minister prime republic europe political fascist william cricketer
Alg 5	pitcher manhattan outfielder boston baseball suffrage millennium prohibition decade ratification
	Topic 32: where can i find growth rates for the pine tree?
Alg 1	Felled tree Taiga Silviculture Christmas tree Maine Everglades National Park
Alg 2	relatively protein spirally shell control internal herbaceous physiology nigra petals
Alg 3	pinus cones wood needles species seeds fir sp pinyon acacia
Alg 4	population theory directly property landau cagr measure produced notation economy
Alg 5	pinus subgenus fir foliage genus

- Many search queries are entered by users in question form. Many of the terms that are added during this process should have an impact on the results which are returned. However, many of the algorithms ignore this.
- Although in many cases, a reference to the original submitted query is used as a stem for the identification of relevant content, no call-back is made in future steps to identify if the terms generated link back to the original user's query.
- Query drift appears in all algorithms, especially in longer queries, as query terms are often expanded independently of the rest of the terms in the search query.
- Shorter search queries often lack context and are not treated with care allowing irrelevant expansion terms to be included in the search process.

6 CONCLUSIONS

This paper described the testing and analysis of five ASQE algorithms on the ClueWeb12 data set using 50 of the TREC-9 Web Topics. To gain an understanding of the existing algorithms for ASQE that use Wikipedia as an external data source for a priori knowledge, each of the five algorithms were recreated using the Python programming language following the specification outlined by the authors. For each algorithm, the tests were performed on the ClueWeb12 data set with a focus on the top 10 results which were returned for each search query. Using these search results, relevance analysis was preformed in the form of AP @10 scores, which utilise the document ranking positions in the results.

A cross-algorithm analysis was performed, outlining how each of the algorithms performed in a side-by-side comparison. As short search queries are often the most difficult for IR engines to interpret due to the lack of context, an analysis of 1 term and 2 term queries was performed, outlining the results of the enhancement process. To gauge an overall view of how each algorithm performed as a whole, the MAP @10 scores using the 50 queries for each algorithm were calculated.

The results from testing and analysis process of Algorithms 1 to 5, outlined that Algorithm 3 was most successful with an overall MAP @10 score of 0.714. Due to the naïve approach taken by Algorithm 2, it performed worst with an overall MAP @ 10 score of 0.384.

The main conclusions that can be drawn from this analysis is that ASQE algorithms that focus on the expansion of queries as a whole and not on individual tokens perform the best as the context of the terms is crucial to their success. Weighting schemes that are utilised no matter how complex or simple, all suffer from the same issue of taking the weights that are generated as the definitive weight for terms without additional post-processing being performed. Each of the algorithms often suffered as they did not make a call-back to the original search query, allowing terms with high weightings to be included in the enhancement process although they are irrelevant to the user's original search intent.

REFERENCES

[1] Bashar Al-Shboul and Sung-Hyon Myaeng. 2014. Wikipedia-based query phrase expansion in patent class search. *Information Retrieval* 17, 5 (2014), 430–451. https://doi.org/10.1007/s10791-013-9233-4

[2] Mohannad ALMasri, Catherine Berrut, and Jean-Pierre Chevallet. 2013. Wikipedia-based Semantic Query Enrichment. In *Proceedings of the Sixth International Workshop on Exploiting Semantic Annotations in Information Retrieval (ESAIR '13)*. ACM, New York, NY, USA, 5–8. https://doi.org/10.1145/2513204.2513209

[3] Avi Arampatzis and Jaap Kamps. 2008. A study of query length. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR '08)*. 811–812.

[4] Christopher Boston, Hui Fang, Sandra Carberry, Hao Wu, and Xitong Liu. 2014. Wikimantic: Toward effective disambiguation and expansion of queries. *Data & Knowledge Engineering* 90 (2014), 22 – 37. https://doi.org/10.1016/j.datak.2013.

07.004 Special Issue on Natural Language Processing and Information Systems (NLDB 2012).

[5] Carson Bruce, Xiaoying Gao, Peter Andreae, and Shahida Jabeen. 2012. *Query Expansion Powered by Wikipedia Hyperlinks.* Springer Berlin Heidelberg, Berlin, Heidelberg, 421–432. https://doi.org/10.1007/978-3-642-35101-3_36

[6] Adam Fourney and Susan T. Dumais. 2016. Automatic Identification and Contextual Reformulation of Implicit System-Related Queries. In *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '16).* ACM, New York, NY, USA, 761–764. https://doi.org/10.1145/2911451.2914701

[7] Maryam Habibi, Parvaz Mahdabi, and Andrei Popescu-Belis. 2016. Question answering in conversations: Query refinement using contextual and semantic information. *Data & Knowledge Engineering* 106 (2016), 38 – 51. https://doi.org/10.1016/j.datak.2016.06.003

[8] Ben He and Iadh Ounis. 2009. *Studying Query Expansion Effectiveness.* Springer Berlin Heidelberg, Berlin, Heidelberg, 611–619. https://doi.org/10.1007/978-3-642-00958-7_57

[9] Yuncheng Jiang, Xiaopei Zhang, Yong Tang, and Ruihua Nie. 2015. Feature-based approaches to semantic similarity assessment of concepts using Wikipedia. *Information Processing & Management* 51, 3 (2015), 215 – 234. https://doi.org/10.1016/j.ipm.2015.01.001

[10] Payam Karisani, Maseud Rahgozar, and Farhad Oroumchian. 2016. A query term re-weighting approach using document similarity. *Information Processing & Management* 52, 3 (2016), 478 – 489. https://doi.org/10.1016/j.ipm.2015.09.002

[11] Khamsum Kinley, Dian Tjondronegoro, Helen Partridge, and Sylvia Edwards. 2012. Relationship between the nature of the search task types and query reformulation behaviour. In *Proceedings of the Seventeenth Australasian Document Computing Symposium.* ACM, 39–46.

[12] Sophie Monchaux, Franck Amadieu, Aline Chevalier, and Claudette MarinÃĈÂĬ. 2015. Query strategies during information searching: Effects of prior domain knowledge and complexity of the information problems to be solved. *Information Processing & Management* 51, 5 (2015), 557 – 569. https://doi.org/10.1016/j.ipm.2015.05.004

[13] Paul Ogilvie, Ellen Voorhees, and Jamie Callan. 2009. On the number of terms used in automatic query expansion. *Information Retrieval* 12, 6 (2009), 666. https://doi.org/10.1007/s10791-009-9104-1

[14] Rishiraj Saha Roy, Smith Agarwal, Niloy Ganguly, and Monojit Choudhury. 2016. Syntactic complexity of Web search queries through the lenses of language models, networks and users. *Information Processing & Management* 52, 5 (2016), 923 – 948. https://doi.org/10.1016/j.ipm.2016.04.002

[15] Bhawani Selvaretnam and Mohammed Belkhatir. 2016. A linguistically driven framework for query expansion via grammatical constituent highlighting and role-based concept weighting. *Information Processing & Management* 52, 2 (2016), 174 – 192. https://doi.org/10.1016/j.ipm.2015.04.002

[16] Pawan Sharma, Rashmi Tripathi, and R.C. Tripathi. 2015. Finding Similar Patents through Semantic Query Expansion. *Procedia Computer Science* 54 (2015), 390 – 395. https://doi.org/10.1016/j.procs.2015.06.045

[17] JesÃžs Vilares, Miguel A. Alonso, Yerai Doval, and Manuel Vilares. 2016. Studying the effect and treatment of misspelled queries in Cross-Language Information Retrieval. *Information Processing & Management* 52, 4 (2016), 646 – 657. https://doi.org/10.1016/j.ipm.2015.12.010

[18] Yang Xu, Gareth J.F. Jones, and Bin Wang. 2009. Query Dependent Pseudo-relevance Feedback Based on Wikipedia. In *Proceedings of the 32Nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '09).* ACM, New York, NY, USA, 59–66. https://doi.org/10.1145/1571941.1571954

[19] Vikrant Yadav and Sandeep Kumar. 2016. Learning Web Queries for Retrieval of Relevant Information About an Entity in a Wikipedia Category. In *Proceedings of the 25th International Conference Companion on World Wide Web (WWW '16 Companion).* International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, Switzerland, 1013–1014. https://doi.org/10.1145/2872518.2891114

[20] D. Zhao, P. Liu, L. Qin, and Y. Li. 2014. A Novel Terms Semantic Query Model Based on Wikipedia. In *2014 11th Web Information System and Application Conference.* 258–261. https://doi.org/10.1109/WISA.2014.54

[21] Gang Zhao, Ji Wu, Dingding Wang, and Tao Li. 2016. Entity disambiguation to Wikipedia using collective ranking. *Information Processing & Management* 52, 6 (2016), 1247 – 1257. https://doi.org/10.1016/j.ipm.2016.06.002

[22] Le Zhao and Jamie Callan. 2012. Automatic Term Mismatch Diagnosis for Selective Query Expansion. In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '12).* ACM, New York, NY, USA, 515–524. https://doi.org/10.1145/2348283.2348354

[23] Meriem Amina Zingla, Latiri Chiraz, and Yahya Slimani. 2016. Short Query Expansion for Microblog Retrieval. *Procedia Computer Science* 96 (2016), 225 – 234. https://doi.org/10.1016/j.procs.2016.08.135 Knowledge-Based and Intelligent Information & Engineering Systems: Proceedings of the 20th International Conference KES-2016.