University of Wollongong

Research Online

University of Wollongong Thesis Collection 1954-2016 | University of Wollongong Thesis Collections

2015

On automatic testing of web search engines

Shaowen Xiang
University of Wollongong

Follow this and additional works at: https://ro.uow.edu.au/theses

University of Wollongong

Copyright Warning

You may print or download ONE copy of this document for the purpose of your own research or study. The University does not authorise you to copy, communicate or otherwise make available electronically to any other person any copyright material contained on this site.

You are reminded of the following: This work is copyright. Apart from any use permitted under the Copyright Act 1968, no part of this work may be reproduced by any process, nor may any other exclusive right be exercised, without the permission of the author. Copyright owners are entitled to take legal action against persons who infringe their copyright. A reproduction of material that is protected by copyright may be a copyright infringement. A court may impose penalties and award damages in relation to offences and infringements relating to copyright material. Higher penalties may apply, and higher damages may be awarded, for offences and infringements involving the conversion of material into digital or electronic form.

Unless otherwise indicated, the views expressed in this thesis are those of the author and do not necessarily represent the views of the University of Wollongong.

Recommended Citation

Xiang, Shaowen, On automatic testing of web search engines, Master of Computer Science thesis, School of Computer Science and Software Engineering, University of Wollongong, 2015. https://ro.uow.edu.au/theses/4310

Research Online is the open access institutional repository for the University of Wollongong. For further information contact the UOW Library: research-pubs@uow.edu.au

University of Wollongong

ON AUTOMATIC TESTING OF WEB SEARCH ENGINES

A Thesis Submitted in Partial Fulfilment of
the Requirements for the Award of the Degree of

Master of Computer Science

from

UNIVERSITY OF WOLLONGONG

by

Shaowen Xiang

School of Computer Science and Software Engineering
Faculty of Engineering and Information Sciences

2015

© Copyright 2015

by

Shaowen Xiang

ALL RIGHTS RESERVED

CERTIFICATION

I, Shaowen Xiang, declare that this thesis, submitted in partial fulfilment of the requirements for the award of Master of Computer Science, in the School of Computer Science and Software Engineering, Faculty of Engineering and Information Sciences, University of Wollongong, is wholly my own work unless otherwise referenced or acknowledged. The document has not been submitted for qualifications at any other academic institution.

Shaowen Xiang
15 Feb 2015

Dedicated to

my wife Yue Liang and my daughter Yanxi Xiang

Table of Contents

List of Tables iv
List of Figures/Illustrations v

ABSTRACT vi
Acknowledgements vii

1 Introduction **1**
1.1 Background 1
1.2 Research Goals 5
1.3 Contributions of the Thesis 6
1.4 Organisation of the Thesis 6

2 Literature Review **8**
2.1 Metamorphic Testing 8
2.1.1 Basic Concepts of Metamorphic Testing 8
2.1.2 The Applications of Metamorphic Testing 11
2.1.3 Constructing and Selecting Metamorphic Relations 12
2.2 Search Engine Evaluation 13
2.2.1 Methods Related to Precision and Recall 13
2.2.2 Methods Related to Ranking Quality 14
2.2.3 Methods Related to Coverage 14
2.2.4 Methods Related to Stability 15
2.2.5 Automatic Evaluation Methods 15
2.2.6 Other Related Literatures on Search Engines 16
2.3 Summary 17

3 Identification of Metamorphic Relations for Search Engines **19**
3.1 Missing Pages 21
3.1.1 MR: MPSite 21
3.1.2 MR: MPTitle 22
3.1.3 MR: MPReverseJD 23
3.2 Swapping Keywords 24
3.2.1 MR: Universal SwapJD 24
3.2.2 MR: SwapJD with Domain 25
3.3 Ranking Drop with Domain 26

3.3.1 MR: Top1Absent . . . 26
3.3.2 MR: Top5Absent . . . 26

4 Empirical Evaluation Using the MRs of No Missing Pages 28
4.1 MR: MPSite . . . 29
4.1.1 Objectives of the Experiment . . . 29
4.1.2 Experimental Design . . . 29
4.1.3 Threats to validity . . . 34
4.1.4 Experimental Results . . . 34
4.2 MR: MPTitle . . . 36
4.2.1 Objectives of the Experiment . . . 36
4.2.2 Experimental Design . . . 36
4.2.3 Threats to Validity . . . 41
4.2.4 Experimental Results . . . 41
4.3 MR: MPReverseJD . . . 44
4.3.1 Objectives of the Experiment . . . 44
4.3.2 Experimental Design . . . 44
4.3.3 Threats to Validity . . . 48
4.3.4 Experimental Results . . . 50

5 Empirical Evaluation Using the MRs of Swapping Keywords 51
5.1 MR: Universal SwapJD . . . 51
5.1.1 Objective of the Experiment . . . 51
5.1.2 Experimental Design . . . 51
5.1.3 Threats to Validity . . . 53
5.1.4 Experimental Results . . . 55
5.2 MR: Swapping Keywords with Domain . . . 58
5.2.1 Objective of the Experiment . . . 58
5.2.2 Experimental Design . . . 58
5.2.3 Threats to Validity . . . 60
5.2.4 Experimental Results . . . 60

6 Empirical Evaluation Using the MRs of No Ranking Drop with Domain 64
6.1 Objective of the Experiment . . . 64
6.2 Experimental Design . . . 64
6.2.1 Independent and Dependent Variables . . . 64
6.2.2 Experimental Procedures . . . 65
6.3 Threats to Validity . . . 66
6.4 Experimental Results . . . 68

7 Additional Findings 70
7.1 Are Search Results Biased by Search Engine for Commercial Interest? . 70
7.2 Correlations between MRs . . . 74

8 Conclusion and Future Work **77**

A Names used in MPReverseJD experiment **79**

B Original queries of No Ranking Drop with Domain **84**

References **95**

List of Tables

3.1 Metamorphic relations defined in this study 20

4.1 The number of test case pairs used to compare the MPSite hourly ROCOF 31
4.2 Multiple comparisons of MPSite hourly ROCOA 37
4.3 The number of test case pairs used to compare MPTitle hourly ROCOA 39
4.4 Multiple comparisons of MPTitle hourly ROCOA 42
4.5 The number of test case pairs in the experiment MPReverseJD 46
4.6 Multiple comparisons of hourly average SRJC 49

5.1 The number of test case pairs in the experiment universal SwapJD . . . 53
5.2 Multiple comparisons of hourly average JCT50 in MR Universal SwapJD 58
5.3 The number of test case pairs in the experiment SwapJD with domain . 59
5.4 Average number of result counts 60
5.5 Multiple comparisons of hourly average JCT50 in SwapJD with Domain 62

6.1 The number of test case pairs in the experiment Ranking Dropping with domain 66
6.2 Multiple comparisons of Top1Absent and Top5Absent hourly ROCOA . 69

List of Figures

4.1 An example of Google MPSite using English query 32
4.2 An example of Google MPSite using Chinese query 33
4.3 MPSite hourly ROCOF of Google, Bing, CBing and Baidu, including English words and Chinese words 35
4.4 An example of Bing MPTitle 40
4.5 MPTitle hourly ROCOA of Google, Bing, CBing and Baidu, including English words and Chinese words 41
4.6 An example of Baidu MPReverseJD 47
4.7 Hourly average SRJC of search engines on three different word categories 48

5.1 An example of Bing SwapJD on 13 January 2014 54
5.2 An example of Bing SwapJD on 27 February 2014 56
5.3 Hourly average JCT50 of Google, Bing, CBing and Baidu 57
5.4 Hourly average JCT50 of search engines in MR SwapJD with Domain . 61

6.1 An example of Bing Top1Absent on 19 January 2015 67
6.2 No ranking Drop with Domain hourly ROCOA of the four search engines 68

7.1 Correlation of MPSite hourly ROCOF and AAPQ of Bing English . . 71
7.2 Correlation between Bing hourly average SRJC and AAPQ 72
7.3 Correlation between hourly average JCT50 and AAPQ 73
7.4 Correlation between Top1Absent and Top5Absent hourly ROCOA of Baidu and AAPQ 73
7.5 Correlation between hourly average JCT50 and Top1Absent hourly ROCOA of Bing 75
7.6 Correlations between MRs of CBing 76

On automatic testing of Web search engines

Shaowen Xiang

A Thesis for Master of Computer Science

School of Computer Science and Software Engineering
University of Wollongong

ABSTRACT

Web search engines are very important because they are the means by which people retrieve information from the World Wide Web. However, testing these web search engines is difficult because there are no test oracles, so this research proposes seven new metrics based on the idea of metamorphic relations to alleviate the oracle problem in search engine testing. Using these metrics, our method can test search engines automatically in the absence of an ideal oracle. Using this method, we further conduct large-scale empirical studies to investigate and compare the qualities of four major search engines, namely, Google (www.google.com), Baidu (www.baidu.com), Bing (www.bing.com), and Chinese Bing (www.bing.com.cn). Our empirical studies involve more than 50 million queries sent to the search engines across 9 months, and about 300 GB data collected from the search engine responses. It is found that different search engines have significantly different performance and that the nature of the query terms can have a significant impact on the performance of the search engines. These empirical study results demonstrate that our method can effectively alleviate the oracle problem in search engine testing, and can help both developers and users to obtain a better understanding of the search engine behaviour under different operational profiles.

KEYWORDS: Software Testing, Oracle Problem, Metamorphic Testing, Reliability, Page Retrieval Capability, Page Ranking Consistency, Search Engines

Acknowledgements

Researching is a hard work, especially for an international student. Without the helps from many people, I would not have been able to complete this research.

Foremost, I would like to express my sincere gratitude to my supervisor Dr. Zhiquan Zhou for the continuous support of my master study and research, for his patience, motivation, enthusiasm, and immense knowledge. His guidance helped me in all the time of research and writing of this thesis.

Secondly, I would like to thank Dr. Markus Hagenbuchner for his helps with data storage devices.

Thirdly, I would like to thank my wife, Yue Liang and my parents, Fucai Xiang and Xiaonie Peng, for their support and encouragement.

Last but not least, I gratefully acknowledge the help of Dr. Madeleine Strong Cincotta in the final language editing of this thesis.

Chapter 1

Introduction

1.1 Background

The goal of software engineering is to develop high quality software whose qualities of correctness and reliability are the most important and fundamental [1]. If a program meets its requirements specification then it is correct; otherwise it is incorrect regardless of the seriousness of the failures. In other words, a program is either correct or incorrect. It is well known that many real-world software products are not correct but they are being used by millions of users every day. This is because some incorrect behaviour is tolerable if they are not serious. That is to say, users could still feel the software is dependable even if it contains some faults. Reliability is a quality that describes this concept [1]. Correctness is an absolute concept whereas reliability is a relative concept. To improve the reliability of a software, it must first be measured. Most reliability metrics involve the identification of failures, that is, if the assessors cannot decide whether the program execution outcomes are correct, then they cannot evaluate the reliability of the software.

To detect failures requires an *oracle*, a mechanism against which a tester can measure the outcomes of program executions and know whether they are correct or not [2]. In some situations, however, an oracle cannot be found or is too expensive to apply. This is known as the *oracle problem* [2], and it is regarded as one of the most difficult problems in software testing [3]. In these situations it is difficult to measure the reliability of *software under test (SUT)*. In this research we considered a very important type of software, namely, Web search engines. Owing to the sheer volume of data on the Internet, there is an oracle problem when testing Web search engines, which meant that evaluating the reliability of Web-based search engines has been very difficult.

Search is the second most popular functionality of the Internet, next to email [4]. The Web search service is one of the most important search services among all the search services, including image search, video search, map search, etc. This study will focus on the Web search service even though the method used in this study also can be used to evaluate other search services as well.

Web search engines such as Google (www.google.com), Bing (www.bing.com), Chinese Bing (www.bing.com.cn, in the rest of this thesis, we will use CBing to represent Chinese Bing) and Baidu (www.baidu.com) allow people to search for information on the World Wide Web. Depending on the queries provided by users, Web search engines (in the rest of this study the phrase 'search engine' refers to 'Web search engine') retrieve all the webpages relevant to these queries and rank them with ranking algorithms. This means the quality of the retrieving and ranking algorithms is responsible for providing high quality search results. In today's highly competitive search market, it is imperative that these search engines provide the desired result according to the queries entered; otherwise, the customers will switch to another search engine. In the context of search engines, the online user manuals of search engines can be regarded as specifications. Therefore, the search engine correctness can be defined as Definition 1.

Definition 1 *(Search Engine Correctness): If a search engine performs in the same way as defined in its user manual, then it is correct, otherwise it is incorrect.*

Thus far it has been considered very difficult to evaluate the reliability of search engines because the metrics used to evaluate traditional software products are hard to apply on search engines. For example, the following five reliability metrics are often used to measure the reliability of software products: [5]

1. MTTF (Mean Time to Failure): Average time between two failures;

2. MTTR (Mean Time to Repair): The average time to locate errors and fix them after failure occurs;

3. MTBF (Mean Time between Failures): This metric is the combination of MTTF and MTTR, which is MTBF=MTTF + MTTR;

4. POFOD (Probability of Failure on Demand): The probability of failure occurring when the software services a request;

5. ROCOF (Rate of Occurrences of Failure): The ratio of total number of failures and the duration of the observation.

These reliability metrics are conventionally hard to apply to search engines because they are all failure-related and it is hard to identify failures from search engine results because there is not enough oracles. However, we find that we can regard online user manuals of search engines as a kind of specifications. Thus, search engine failure can be defined as Definition 2.

Definition 2 *(Search Engine Failure): Search engine failure is the incapacity of a search engine to conduct its required functions according to its user manual. A search engine failure occurs if the behaviour of the search engine is different from the specified behaviour.*

In the present study, we discover some logical consistency properties using the online user manuals of search engines. Because these consistency properties are carefully designed based on search engines' user manuals, if search engines violate them we can consider there are real failures or anomalies in search engines. We then apply the metamorphic testing method to alleviate the oracle problem in search engine testing, so that we can detect failures or anomalies in search engines and then these metrics are able to apply on them. For instance, our method can detect a type of failures that the keyword "site" does not work as specified in search engines' user manual (This type of failures will be described in detail in Section 3.1.1). We choose ROCOF to measure the reliability of search engines because MTTF, MTTR and MTBF require real time monitoring over a long period, whereas our experiments were conducted by sampling. To measure the occurrences of anomaly, a new metric *ROCOA* is introduced and defined as Definition 3.

Definition 3 *(ROCOA): ROCOA is the Rate of Occurrences of Anomalies that is the ratio of total number of anomalies and the duration of the observation.*

The quality metrics of traditional information retrieval systems are also difficult to be used in the evaluation of search engines because a Web search engine is a special information retrieval system designed to retrieve information from the World Wide Web. Cleverdon et al. proposed the use of six metrics, coverage, time lag, recall, precision, presentation and user effort to evaluate an information retrieval system [6]. However, search engines differ from the traditional information retrieval systems, which makes some of these conventional metrics hard to apply. For instance, recall and precision are not suitable for search engines. When sending a query to a search engine, if A is the set of all the results returned by the search engine, and then we just suppose R, a subset of A, is the set of all the relevant results to the query while R' is the set of relevant results that were not retrieved, then the precision is calculated as $|R| \div |A|$ and the recall can be calculated as $|R| \div (|R| + |R'|)$ [7]. Obviously, these two metrics cannot be calculated because R' is unknown and it is difficult to distinguish relevant results from irrelevant results since different users' view of relevance are different, so new and more appropriate metrics are needed to evaluate search engines [8].

Bar-Ilan et al. [9] monitored five queries (three text queries and two image queries) over a period of about three weeks to study the stability (equal ranking) of each individual search engine during that period. They found that Google had the most stable result set and result rankings during that period.

Zhou et al. [7, 10] pointed out that the *logical consistency* relationships among *multiple* responses can be used to measure search engine qualities in the absence of an oracle. These logical consistency relationships among multiple responses are known as *metamorphic relations* in *metamorphic testing* [11] and therefore Zhou et al.'s method is an application of metamorphic testing.

One of the most frequently discussed qualities of search engines is their "semantic" ability, which indicates their accuracy of understanding the contextual meaning of terms. Imielinski and Signorini [12] argued that a truly semantic search engine should be insensitive to semantically equivalent rephrases. For example, the search result page for "capital of France" and "which city is France's capital" should both contain the answer "Paris". This testing method of using semantically equivalent rephrases also belongs to the category of metamorphic testing [7, 10, 11] because it employs the logical consistency relationships among multiple responses of the search engine under test.

Following the idea of metamorphic testing, in this study we develop seven metrics suitable for search engine evaluation, with a focus on the retrieving capability and ranking ability of the search engines under different operational profiles. From the perspective of software quality assessment, operational profiles are needed since different users may use the search engine in different ways.

1.2 Research Goals

This research has two main goals:

I. To develop methods of alleviating the oracle problem when assessing the page retrieval capability and the page ranking consistency of Web search engines using the concept of metamorphic testing.

II. To conduct empirical evaluations using major Web search engines such as Google, Bing, CBing and Baidu.

1.3 Contributions of the Thesis

1. This thesis applied the metamorphic testing method to alleviate the oracle problem in search engine testing.

2. This thesis proposed seven MRs which can be used to evaluate the page retrieval capability and the page ranking consistency of search engines.

3. This thesis conducted experiments using the seven MRs to empirically evaluate four commercial search engines for nine months. A comparison of the page retrieval capability and the page ranking consistency of these four search engines was made.

4. This thesis analysed the correlations between the page retrieval capability and the page ranking consistency of the search engines and some other factors such as advertisements in the result pages and query languages. This information is useful for both users and developers to understand the behaviour of the search engines, and provides hints for debugging and tuning the search engines.

5. This thesis also analysed the correlations between the anomaly-detection effectiveness of different metamorphic relations.

1.4 Organisation of the Thesis

The remainder of this thesis is laid out as follows.

In Chapter 2, literature on metamorphic testing and evaluating search engines is reviewed. Chapter 3 identifies seven *metamorphic relations* used to evaluate search engines and then categorises them into three categories: missing pages, swapping keywords, and no ranking drop with domain. In Chapters 4, 5, and 6 empirical studies using the three categories of MRs are conducted to evaluate the search engines. Chapter 7 analyses the correlation between the seven metrics proposed in Chapter 3, and Chapter 8 presents the conclusion of this thesis and suggestions for future research.

Chapter 2

Literature Review

2.1 Metamorphic Testing

2.1.1 Basic Concepts of Metamorphic Testing

Software testing normally includes three steps: 1. select test cases of the SUT; 2. execute the test cases; and 3. verify the outputs of the test cases. Test cases which the SUT has computed correctly are called *successful test cases*, but testers are often consider them to be less useful and ignore them because they do not reveal any failures [13].

Chen et al., however, found that the information carried by successful test cases is also valuable [13], but the question of how to effectively utilise these successful test cases is an important topic in software testing [14], because testing is still very expensive and accounts for a major part of the total development cost [15]. This means that successful test cases must be used efficiently, which is why fault based testing makes the best of every test case because it uses successful test cases to prove the absence of certain types of faults [16, 17].

Based on the idea of making use of successful test cases, Chen et al. proposed *metamorphic testing (MT)* to alleviate the oracle problem. MT uses consistency properties, which are *metamorphic relations (MRs)*, to generate test cases and then verify the results; this makes it possible to test a program without an oracle. For example, we want to test a software that computes the sine function, so given a test case of say 55.5 where the corresponding result is 0.824. There is no oracle to judge whether the output of the program is correct or not, but here the property sin (x) = sin (360 + x) can be used as a metamorphic relation. We can also derive a follow-up query 360 + 55.5 = 415.5 and send it to the program and if we suppose an output of sin(415.5) = 0.818, we can then determine there is a failure in the program because the results do not satisfy the MR. [13]

MT is typically conducted in the following four steps: [18–20]

(1) Identify MRs. This step needs specific domain knowledge of the SUT so the tester should first discuss the properties with a specialist.

(2) Select *original test cases* and execute them.

(3) Generate *follow-up test cases* according to the original test cases and the MRs, and then execute them.

(4) Verify the outputs of the original and the follow-up test cases against the corresponding MRs. If the SUT computed the test cases correctly, the outputs of the original and follow-up test cases should abide by the corresponding MRs [19].

Wu [21] proposed an enhanced version of metamorphic testing by applying a chain of MRs, namely n-iterative metamorphic testing. The author argued that this new version MT can utilise more information than traditional ones. Two algorithms of n-iterative MT with different inputs were introduced, with one working on an MR sequence and the other on a set of MRs. A comparison of the effectiveness of n-iterative metamorphic testing and that of other testing methods has been made and finally revealed that the n-iterative MT method outperformed metamorphic testing and special case testing in terms of generating test cases and finding faults.

Liu et al. [22] conducted an empirical study to show that metamorphic testing is easy to understand and use. They selected five Java programs as the subjects of their experiment. These five Java programs were neither too complex nor too simple so that they are not hard to understand. They recruited university students without the knowledge of metamorphic testing as testers and give them three hours training of metamorphic testing and the target programs. Then the testers developed MRs individually and these MRs are used to test the target programs. The result showed that the collection of all these MRs can be as efficient as a test oracle. They also pointed that the more complex programs need more MRs to be as efficient as a test oracle. The settings of this research is different from Liu et al.'s work in that the latter used controlled experiments where the faults in the subject programs are known in advance, whereas our research uses real search engines where the defects or problems are unknown. Therefore, in this research we do not intend to compare the effectiveness of MRs against that of a real oracle, as a real oracle is not available at all for Web search engines.

Cao et al. [23] conducted empirical study to analysis the correlation between the fault-detection effectiveness of MRs and the dissimilarity (distance) of test case execution profiles which records some aspects of a program's execution. The results showed that the branch-based metrics and the the fault-detection effectiveness of MRs have strong correlation. They showed that their findings can be used to prioritise MRs for cost-effective metamorphic testing.

Similarly, Chen et al. [24] propose a cost-driven approach for metamorphic testing by designing metamorphic relations sharing the same test inputs to reduce the testing cost. They also conducted experiment to show that MRs constructed by their approach are more cost-effective than MRs constructed by traditional approach.

In order to reduce the human effort in constructing MRs, Kanewala [25] proposed an approach to automatically predicting MRs using machine learning techniques. He used extracted features and graph kernels to develop machine learning prediction models to predict MRs of a new function. Their preliminary results showed that their approach is highly effective in predicting metamorphic relations.

2.1.2 The Applications of Metamorphic Testing

Since the arising of MT, it has been widely used to test various programs from a variety of disciplines.

In [20] and [26], MT was applied to test bio-informatics programs. In [20], Chen et al. applied MT to test a network simulator as well as a short mapping program. The authors pointed out that MT is a simple but effective method to test bio-informatics programs. A similar study by Sadi et al. [26] applied MT to test mutant versions of three phylogenetic inference programs, with the results showing that MT could automatically test this kind of program. The authors also found that different MRs fit different mutants so it is better to identify a variety of MRs to test a program.

Xie et al. [27] conducted an empirical study to show the effectiveness of MT in machine learning classifiers by applying MT to Weka 3.5.7, an open-source machine learning package. The authors detected real faults of this popular open-source software using only simple MRs which do not require deep domain knowledge. Thus the authors found that MT could effectively test these classification algorithms.

Yao et al. [28] employed MT in detecting invisible integer bug which is one of the main reasons that cause software calculation error. Their result proved that this MT based method is validated to find hidden integer bugs.

MT was also applied to many other fields such as testing image processing operations [29, 30], testing context-sensitive middle ware application [31], and analysing the feature model [32].

2.1.3 Constructing and Selecting Metamorphic Relations

The MR identification is of great importance in the process of MT because we can save both time and resources if we can identify MRs with high effectiveness.

Liu [18] proposed a formal methodology for systematically identifying metamorphic relations where new MRs are automatically constructed at low cost, based on the original MRs. This method can save a great deal of human effort in identifying MRs.

Normally, many MRs can be identified for one SUT, of which some are highly effective whereas others are not. Therefore, general rules are required to evaluate the effectiveness of MRs so that effective MRs can be selected.

Asrafi et al. [19] conducted a case study aimed at systematically investigating the relationship between the effectiveness of MRs and the code coverage achieved by them. Their results showed that MRs with low code coverage were very ineffective at detecting faults, while MRs with high coverage were in most cases very effective. The authors also pointed out that a certain number of MRs with high coverage could not detect all the faults because these MRs could not achieve full code coverage.

Mayer and Guderlei [33] conducted an empirical study with several Java programs of determinant computation using some metamorphic relations to evaluate the usefulness of MT. They found that MRs that contained much the same semantics as the SUT were normally very effective at detecting failures, whereas those with the form of equalities were very weak. They also pointed out that testers should not use the MRs that are close to the strategy of the typical implementation algorithm.

2.2 Search Engine Evaluation

As discussed in section 1.1, search engines suffer from the oracle problem, which makes it difficult to evaluate their quality, which is why many researchers have tried to develop reasonable methods to evaluate the quality of search engines [7].

2.2.1 Methods Related to Precision and Recall

As stated in section 1.1, the precision and recall cannot be applied directly onto the live Web, so some studies used a modified precision (precision of top 20) and a modified recall (relative recall) to evaluate the search engines.

Hawking et al. calculated the precision of the top 20 results of four popular commercial Web search engines (plus one research system) and compared those results with the results of six Text Retrieval Conference (TREC) systems [34]. They stated that the six TREC systems performed better than the commercial Web search engines and the problem experienced by the commercial Web search engines may stem from the retrieving algorithm rather than the ranking algorithm.

Clarke and Willett [35] stated that it is better to use relative recall rather than absolute recall to evaluate search engines such as AltaVista, Excite, and Lycos. They gathered all the relevant pages returned by different search engines together as a relevant document pool to calculate the relevant recall.

The above studies used a modified form of the conventional measurements of recall and precision to evaluate the search engines. To evaluate the precision of the top 20 and relative recall needs human judgment of relevance, but since relevance is a highly debatable term, to a certain extent the results are unavoidably subjective.

2.2.2 Methods Related to Ranking Quality

It is also very important for search engines to rank the results pages retrieved by them because while they often return a large number of results, most people only visit the top 50, or even the top 20 results. Therefore, it is important to include the most relevant results in the highest ranking. Previous research has evaluated the ranking quality of search results based primarily on human judgment.

Su [36] studied search engines' rankings using human judgment. 36 users were asked to manually select and rank the five most relevant results from the first 20 results returned by three search engines, and then the similarity between the human ranking and the ranking of three search engines was analysed. The result revealed that the similarity between users and the search engines' rankings was low.

Similarly, Bar-Ilan et al. asked 67 students to identify and rank the top 10 results from all results returned by three search engines (Google, MSN Search, and Yahoo!) in [37]. Their aim was to investigate the similarities between human ranking and search engines ranking. They also found that the correlation between the two rankings was low.

2.2.3 Methods Related to Coverage

Some other studies use coverage as the metric to evaluate search engines. For instance, Lawrence and Giles studied six search engines (HotBot, Lycos, AltaVista, Northern Light, Excite and InfoSeek) and found that their coverage of the Web varied substantially [38]. They also revealed that all the six search engines only covered less than about one third of the Web. HotBot covered 34% of the Web, which was the highest coverage while Lycos had the lowest coverage of 3%. The other four search engines AltaVista, Northern Light, Excite and InfoSeek had coverage between these two extremes.

Vaughan and Thelwall [39] tested three search engines (Google, AllTheWeb and AltaVista) for national biases in the coverage of commercial Web sites. The result showed that the three search engines had significant differences in the coverage of commercial Web sites. They pointed out that the sites from the US were much better covered than sites from the other places in the study.

The metric coverage can only indicate which search engine covers the larger portion of the Web, however it does not show the reliability of the search engines.

2.2.4 Methods Related to Stability

Some studies evaluated the stability of search engines in terms of search results over a certain period of time. The stability of the search results meant that the results returned by the search engines remained the same over a period of time.

The query "cataloging department" was sent to Google once a week by Zhao to check the stability of Google [40]. The experiment last for ten weeks and the changes in the ranks of the 24 sites among the top 20 pages were monitored during this period. 21 out of 24 Web sites changed their position at least once.

Vaughan [41] proposed a set of three measurements to evaluate the stability of search engines. These measurements were: (1) the stability of the result count; (2) the overlap of the top 20 results of the two tests; and (3) the ranking of the top 20 results remaining the same between the two tests. The results showed that Google was the most stable of the three search engines and Teoma's was the worst.

2.2.5 Automatic Evaluation Methods

The study by Soboroff et al. [42] examined the rankings of search results without any users' judgment. They based their study on the findings by that a little overlap in the human judgments of relevance would not affect the relative performance evaluated by the different systems. They proposed a ranking system using a number of randomly selecting “pseudo-relevant” documents, but Aslam and Savell observed that Soboroff et al.’s method was not good at predicting the performance of the top performing systems [43].

Can et al. [44] presented an automatic method for evaluating the Web search engines, and they argued that it was an efficient and effective tool for assessing Web search systems. They experimented on eight Web search engines, including AllTheWeb, AltaVista, Hot-Bot, InfoSeek, Lycos, MSN, Netscape, and Yahoo!, by using 25 queries. The researchers used binary user relevance judgments to judge the top 20 results. The result showed that their method provided results which were statistically consistent with human based methods.

Zheng et al. [45] mined rules between a set of items of search results as pseudo test oracles. They proposed three kinds of rules: (1) implications between Websites, (2) the different opinions of search engines about certain Websites and (3) the best top one result of queries. These rules can be used to automate the evaluation of search engines.

Zhou et al. proposed the concept of using logical consistency (that is, metamorphic relation) among multiple responses to test search engines in [7]. Using the concept of metamorphic testing [46, 47], many metrics can be developed.

Zhou et al.’s work was from the perspective of functional testing (that is, testing search engines for functional correctness). In this study we develop new metrics to evaluate search engines using the concept of metamorphic relations.

2.2.6 Other Related Literatures on Search Engines

Altingovde et al. [48] studied the “no-answer” queries and hard queries that retrieved few results using three search engines (Bing, Google and Yahoo!). They pointed out that it was beneficial to characterise and solve no-answer queries so they analysed the ways different search engines corrected no-answer queries and found that they used four patterns to deal with queries with few results. They also found that all the three search engines tried to correct most of the hard queries. Search engine A (not named by the authors) directly provided the suggested query results for about 62% of the hard queries, while search engines B and C provided a query suggestion for most of the hard queries. They argued there was some room for improvement because some hard queries still had no answers.

Long et al. [49] evaluated three Chinese commercial search engines based on human judgments. The three search engines were Google China (http://www.google.com/intl/zh-CN), Yahoo China (http://www.yahoo.cn/) and Baidu (http://www.baidu.com). They investigated the factors that affected the performance of the search engines by monitoring the overlap on the first results page of these three search engines and then calculated the correlation of the search results page and the result page content. The results showed that Spearman's rho coefficient correlation between search results page and result page content of the three search engines were 0.357, 0.360 and 0.385 with $p<0.001$ for Baidu, Google China and Yahoo China, respectively.

Some researchers studied the sponsored links of search engines. Jansen compared the relevance ratings of sponsored links and non-sponsored links in [50] and showed that the relevance ratings of the two kinds of links were slightly different.

2.3 Summary

This chapter reviewed the literature on MT and search engine evaluation. Metamorphic testing can be used in situations where there is either no test oracle or very few, therefore the present study will apply MT to test the search engines.

Almost all the works cited on the evaluation of search engines did not evaluate the reliability of search engines using an operational profile, which assumes that all the users will only use search engines in one way. In reality, users use search engines in different ways, for example some will use different languages and some others are interested in results form specific domains. The present study used different usage patterns to conduct empirical evaluations from the perspective of reliability. These different usage patterns included different query languages, different domains, queries of different semantic meanings and queries of different potential commercial value, to name a few.

Chapter 3

Identification of Metamorphic Relations for Search Engines

Seven metamorphic relations that are useful to evaluate search engines are proposed in this section. As Table 3.1 indicates, the seven metamorphic relations are MPSite, MPTitle, MPReverseJD, Universal SwapJD, SwapJD with Domain, Top1Absent and Top5Absent, and they belong to three groups.

In this table, the metric for MR MPReverseJD is *Search Result Jaccard Coefficient (SRJC)* which is defined as the cardinality of the intersection of the original query result set and the follow-up query result set divided by the cardinality of the union of the tow sets. The SRJC can be given by Equation 3.1.

$$SRJC = \frac{|\{original\_query\_results\} \cap \{follow\_up\_query\_results\}|}{|\{original\_query\_results\} \cup \{follow\_up\_query\_results\}|} \tag{3.1}$$

To measure SwapJD, we calculate the Jaccard coefficient of top 50 results of original query and top 50 results of follow-up query. In this thesis, we denote the top 50 results of the original query results as *OQ50* and the top 50 results of the follow-up query

Table 3.1: Metamorphic relations defined in this study

Group	Name	Usage pattern	Result of each single metamorphic test	Result of each batch of test	Frequency of test	Do different batches use the same test suite?	Goal
No Missing Page	MPSite	English Chinese	{pass, fail}	Hourly ROCOF [0.0, 1.0]	1 batch per hour	No	To test the search engine's page retrieval capability, focusing on its reliability of retrieving pages that contain an exact word or phrase.
	MPTitle	English Chinese	{found, not found}	Hourly ROCOA [0.0, 1.0]	1 batch per hour	No	To test the search engine's page retrieval capability, focusing on its capability of abstracting a page and understanding user intent.
	MPReverseJD	Persons' names Company names Drug names	SRJC [0.0, 1.0]	Hourly average SRJC [0.0, 1.0]	1 batch per hour	No	To test the search engine's page retrieval capability, focusing on its stability for similar queries that only differ in word order.
Swapping Keywords	Universal SwapJD	Universal	JCT50 [0.0, 1.0]	Hourly average JCT50 [0.0, 1.0]	1 batch per hour	Yes	To test the search engine's consistency in page ranking, focusing on its stability for similar queries that only differ in word order.
	SwapJD with Domain	site:com site:edu site:mil site:lc					
No Ranking Dropping with Domain	Top1Absent Top5Absent	Random English words	{dropped, not dropped}	Hourly ROCOA [0/500, 500/500]	1 batch per hour	Yes	To test the search engine's consistency in page ranking, focusing on its consistency with different domains.

results as *FQ50*. Then the metric *Jaccard Coefficient of top 50 results (JCT50)* is defined as:

$$JCT50 = \frac{|\{OQ50\} \cap \{FQ50\}|}{|\{OQ50\} \cup \{FQ50\}|} \tag{3.2}$$

3.1 Missing Pages

This group of metamorphic relations is designed to test search engines' page retrieval capability, which is to test whether or not there are any search results missing from the search engines' search results. In this thesis, all the advertisement results are removed from the search results.

3.1.1 MR: MPSite

This metamorphic relation is designed to test the search engine's page retrieval capability, focusing on its reliability of retrieving pages that contain an exact word or phrase. In the present thesis, only English words and Chinese words are used to query search engines and the "word" is defined in Definition 4.

Definition 4 *(Word): An English word is an entry of an English dictionary with 127,141 entries, which is downloaded from "Oracle" website [51], while a Chinese word is a single Chinese character from a dictionary with 10,000 entries, which was collected by the author.*

In the english dictionary, some words may have spelling mistakes, which is appropriate in the experiments in this thesis because real users often make some spelling mistakes when they are typing queries to search engines.

Original query: Randomly select a query *"A"* (with quotes) which has a less than 20 result count. In the present thesis, a query may includes one or more words.

Follow-up queries: "A" (with quotes) + site:[the top level domain name of each result of the original query], the i^{th} follow-up query is the one added the domain name of the i^{th} result of the original query.

Verification: If the i^{th} $(0 < i < 21)$ follow-up query does not retrieve the i^{th} result of the original query, then a failure has been detected.

In this experiment, quotation marks will always be used to bracket the query "A". According to the manual page of the four search engines, search engines will find results that include exact the words inside quotes [52–55]; otherwise, some similar results may also be included, so quotation marks will always be used to bracket the query "A". The reason for using quotation marks to bracket queries in all other MRs is the same as the above reason. The result of a single test is either pass or fail. If a test case does not satisfy this consistency property, then the test case finds a failure in the search engine. The score of one batch of tests is the failure rate (ROCOF) in that batch. A batch of test cases was tested every hour, but the test cases in different batches were not necessarily the same.

3.1.2 MR: MPTitle

The aim of this metamorphic relation is to test the search engine's page retrieval capability, focusing on its capability of abstracting a page and understanding user intent.

Original query: Randomly select a query *"A"* (with quotes) which has a less than 20 result count.

Follow-up queries: "A" (with quotes) + [the title of each result of the original query], the i^{th} follow-up query is the one added the title of the i^{th} result of the original query.

Verification: If the i^{th} $(0 < i < 21)$ follow-up query does not retrieve the i^{th} result of the original query, then a anomaly has been detected.

In this experiment, quotation marks will always be used to bracket the query "A", but no quotation will be used to bracket the title of query results. This is because the title is a description generated by the search engine rather than a string directly copied from the target Web page. Therefore, double quotes should not be applied. As a result, the search engine's semantic search capability is tested. The result of a single test is either "found" or "not found". The score of one batch of tests is the anomaly rate (ROCOA) in that batch. A batch of test cases was tested every hour. Different batches contain different test cases (queries).

3.1.3 MR: MPReverseJD

This MR is designed to test the search engine's page retrieval capability, focusing on its insensitivity to similar queries that only differ in word order.

Original query: "A_1" + "A_2" [+ "A_3"] [+ "A_4"] (with quotes), where A_1, A_2, A_3 and A_4 may include one or more words (The brackets in this expression indicate that the contents inside them are optional. That is, A_3 and A_4 are optional in this experiment, if A_1+ A_2 has less than 20 result count, then the remaining words are not needed and therefore the query may include 2 to 4 words)

Follow-up query: ["A_4" +]["A_3" +] "A_2" + "A_1" (with quotes).

Verification: To what extent are the results of the original query and the follow-up query in common?

In this experiment, quotation marks will always be used to bracket the query "A_i". The result of a single test is a Jaccard coefficient between the result set of the original query and the result set of the follow-up query, which is between 0.0 and 1.0. The score of one batch of tests is the average Jaccard coefficient of the test cases in the batch. A batch of tests were tested every hour, but the test cases in different batches were not necessarily the same. The higher value of one single test means that the search engine is less sensitive to similar queries that only differ in word order. Because words in a query were selected from only one word category and are all names, so that the semantic of the reversed order query was similar to the original query and the keywords in the two queries were the same. On this basis it was reasonable to believe they would return a large number of common results, if the search engine were in good quality. If there is any difference between the two result sets, then it means either the original query do not retrieve all the relevant results or the follow-up query do not retrieve all the relevant results. Therefore, from the perspective of users, we can expect this value to be high.

A_i ($i \in N$, $0 < i < 5$) in a query were selected from only one word category and are all names, so that the semantic of the reversed order query was similar to the original query and the keywords in the two queries were the same, so it was reasonable to believe they will return a large number of common results if the search engine is stable.

3.2 Swapping Keywords

These MRs are designed to test the search engine's consistency in page ranking, focusing on its insensitivity to similar queries that only differ in word order. Although these MRs use the concept of Jaccard coefficient similarly as used in MPReverseJD, these MRs do not restrict to those query who has only 20 results. That is to say, in these MRs, a query may has millions results returned, but we only focus on the first 50 results.

3.2.1 MR: Universal SwapJD

Original query: $A + B$, where A and B are words without quotes.

Follow-up query: $B + A$

Verification: To what extent are the results of the original query and those of the follow-up query the same?

In this experiment, no quotation marks will be used to bracket the queries, because we do not want to search exact phrases but the semantic meaning search. The result of a single test is JCT50 which is between 0.0 and 1.0. If the result was too low (depending on a value given by the user), then there is an anomaly. The value of one single test indicates the seriousness of the anomaly. The score of one batch of tests is the average JCT50 of the test cases in the batch. A batch of tests was tested every hour and the test cases in different batches were the same.

3.2.2 MR: SwapJD with Domain

These MRs are tested separately to measure the search engine's consistency in different domains, so that we can understand the effect of the domain scale on the performance of a search engine.

Original query: $A + B +$ site:[one of these four domain name: ".com", ".edu", ".mil" and ".lc"]

Follow-up query: $B + A +$ site:[the same domain name as the original query]

Verification: To what extent are the results of the original query and those of the follow-up query the same?

In this experiment, no quotation marks will be used to bracket the queries, because we do not want to search exact phrases but the semantic meaning search. The result of a single test is JCT50 which is between 0.0 and 1.0. If the result was too low, then there is an anomaly. The value of one single test indicates the seriousness of the anomaly. The score of one batch of tests is the average JCT50 of the test cases in the batch. A batch of tests was tested every hour and the test cases in different batches were the same.

3.3 Ranking Drop with Domain

These two MRs are to test the search engine's consistency in page ranking, focusing on its consistency with different domains.

3.3.1 MR: Top1Absent

Original query: Randomly select a query "A" (with quotes) from an English dictionary [51].

Follow-up query: "A" (with quotes) + site:[the top level domain name of the first result of the original query].

Verification: If the top 50 results of the follow-up query do not include the first result of the original query, then an anomaly has been detected.

In this experiment, quotation marks will always be used to bracket the query "A_i". The result of a single test is that the ranking dropped or not dropped. All the advertisement results are removed from the search results and all reported anomalies are repeatable at the time of the experiment. Therefore, the anomaly is not owing to data updates. It is to be noted, however, that an anomaly does not necessarily imply a failure, but does imply that the search results are unexpected and hence the search engine developer should look into the anomalies to identify potential faults if any. The score of one batch of tests is the drop rate (ROCOA) of the test cases in the batch. A batch of tests was tested every hours, and the test cases in different batches were all the same.

3.3.2 MR: Top5Absent

Original query: Randomly select a query "A" (with quotes) from a dictionary.

Follow-up queries: "A" (with quotes) + site:[the top level domain name of the top five results of original query], the i^{th} ($i \in N$, $0 < i < 6$) follow-up query is the one added the domain name of the i^{th} result of the original query.

Verification: If the top 50 results of the i^{th} ($i \in N$, $0 < i < 6$) follow-up query do not include the i^{th} result of the original query, then an anomaly has been detected.

In this experiment, quotation marks will always be used to bracket the query "A". Obviously, the Top1Absent is a special case of the Top5Absent when i is equal to one, therefore, in this study we do one trail experiment to analyse the two metrics together. Other characteristics of this metamorphic relation are the same as those of the Top1Absent.

Chapter 4

Empirical Evaluation Using the MRs of No Missing Pages

To obtain the most accurate results, several search settings were considered before this experiment. The SafeSearch was turned off so that it would not filter any content from the search results. Because some search engines may return more relevant results and recommendations based on users' search activities when users are signed in, all accounts were signed out during testing. Also, the search engines may omit some entries that are very similar to the results already displayed, which may lead to an inaccurate result. For this reason, this filter was also turned off. In the rest of this thesis, all the experiments use the same search engine setting as listed above. In this study, IBM SPSS Statistics will be used to analyse test data.

4.1 MR: MPSite

4.1.1 Objectives of the Experiment

This experiment is designed to test search engines' page retrieval capability, focusing on their reliability of retrieving pages that contain an exact word or phrase. Four search engines were included in this experiment, including Google (www.google.com), Baidu (www.baidu.com), Bing (www.bing.com) and CBing (www.bing.com.cn). We also compare the differences between each search engine when English and Chinese queries are used.

4.1.2 Experimental Design

4.1.2.1 Independent and Dependent Variables

The independent and dependent variables of this experiment are listed below:

Independent variables: language (English and Chinese), search engines (Google, Bing, CBing and Baidu)

Dependent variable: MPSite hourly ROCOF.

According to the independent variables, we have eight scenarios (operational definitions), namely Google English, Bing English, CBing English, Baidu English, Google Chinese, Bing Chinese, CBing Chinese, and Baidu Chinese. The experiment tested the MPSite of each of the eight scenarios and compared their MPSite hourly ROCOF.

4.1.2.2 Experimental Procedures

The original query and follow-up query in the experiment were defined as:

Original query: Randomly select a query *"A"* (with quotes) with fewer than 20 results. The way to come up with a query with fewer than 20 results is described as follows. First select one word from one of the dictionaries mentioned in Section 3.1.1.

If the there are more than 20 results, then we add one more word to the query and try again. At most four words are included in a query. If there are already four words in the query but the result count still larger than 20, we start to select a new word from the dictionary as a new query. Then we repeat the above steps until the result count is less than 20.

Follow-up queries: *"A"* (with quotes) + site:[the top-level domain name of each result of the original query], the i^{th} follow-up query is the one added the domain name of the i^{th} result of the original query.

The reason for selecting a query with fewer than 20 results is because it is easy to record all the results and see whether they will appear in the results of their corresponding follow-up queries.

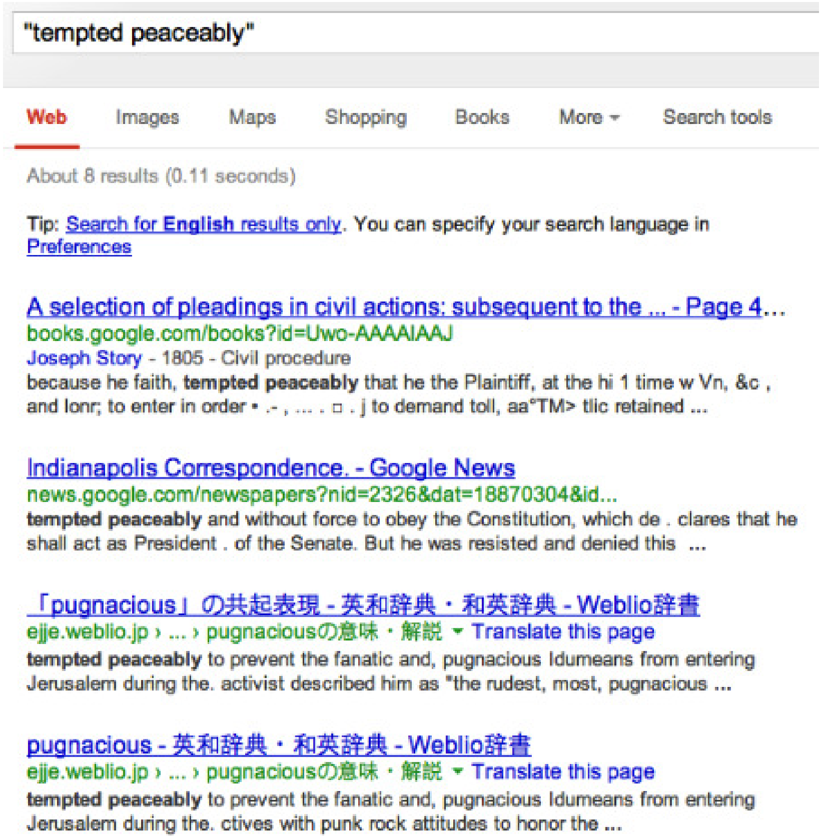
Figure 4.1 is an example of this experiment using English query and Figure 4.2 is an example using Chinese query. In Figure 4.1, the first result of original query is missing after adding "site:.com". Similarly, in Figure 4.2, the first result of original query is missing after adding "site:.au". In the example of English query, the way how the MPSite ROCOF is calculated is described below. Since the original query "tempted peaceably" has eight results (this can be seen from the result count), by adding the domain names of these eight results to the original query we can get eight follow-up queries. Each of these eight follow-up queries and the original query consist of a *test case pair*. In this example, the first four test case pairs are ("tempted peaceably", "tempted peaceably" site:.com), ("tempted peaceably", "tempted peaceably" site:.com), ("tempted peaceably", "tempted peaceably" site:.jp) and ("tempted peaceably", "tempted peaceably" site:.jp). Figure 4.1 shows that the first test case pair detected a failure because the follow-up query " 'tempted peaceably' site:.com" did not retrieve the first result of the original query even though it did also belong to domain ".com". Therefore, one failure was found by these eight test case pairs, then the MPSite ROCOF is 0.125.

In this experiment, about 3000 test case pairs were tested every hour. The MPSite hourly ROCOF is calculated as the number failures in an hour divided by the total number of test case pairs tested in that hour.

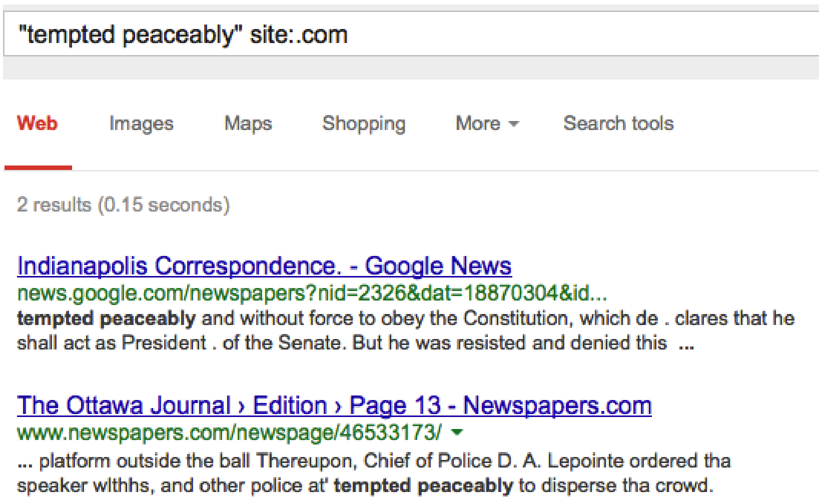
This experiment was conducted to evaluate the MPSite hourly ROCOF of different scenarios. Whenever a test query needs to be issued, query words would be randomly selected from an English (or Chinese) dictionary. New words are added to the query until the result count becomes smaller than or equal to 20. As a result, for each search engine under test, different queries were issued at different times. Table 4.1 shows the number of test case pairs which were used to compare the MPSite hourly ROCOF of different scenarios. These numbers might differ from the numbers of test case pairs used to analyse correlations, because only the results of those hours when all the eight scenarios were tested were used to compare the MPSite hourly ROCOF of different scenarios. This also fits the rest of this study. According to the table, 379 hours data were used to compare the MPSite hourly ROCOF of the eight different scenarios and the total number of test case pairs was about 7,580,000. Because all the test cases used in every hour were randomly selected, it is infeasible to include the search set in this thesis.

Table 4.1: The number of test case pairs used to compare the MPSite hourly ROCOF

Search Engine	Usage Pattern	Test case pairs per hour (approximate)	Hours	Total test case pairs (approximate)
Google	English	1000	379	379,000
	Chinese	1000	379	379,000
Bing	English	3000	379	1,137,000
	Chinese	3000	379	1,137,000
CBing	English	3000	379	1,137,000
	Chinese	3000	379	1,137,000
Baidu	English	3000	379	1,137,000
	Chinese	3000	379	1,137,000

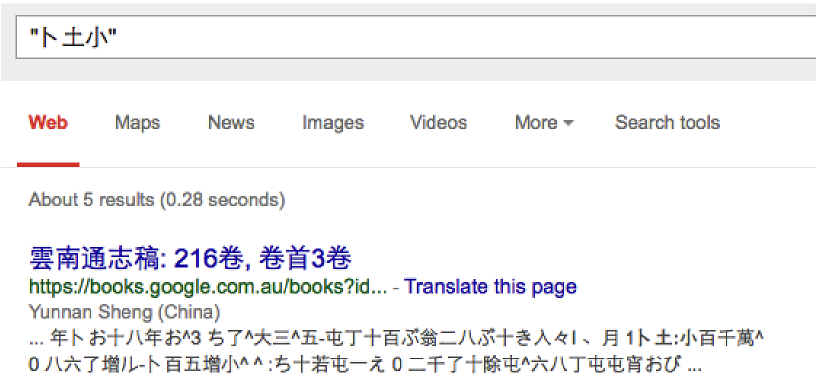


(a) Original query

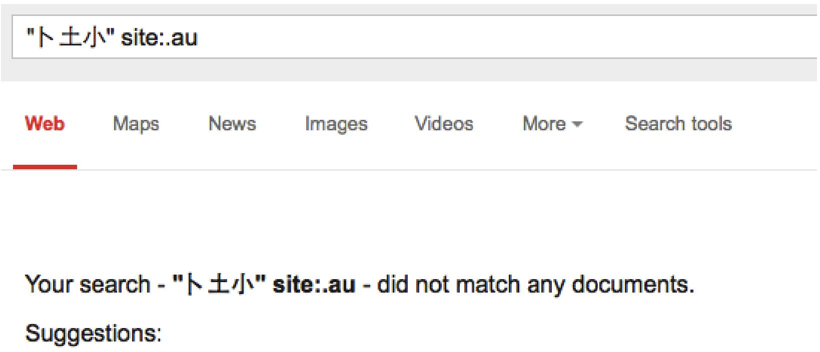


(b) Follow-up query

Figure 4.1: An example of Google MPSite using English query: the first result of original query is missing after adding site:.com (a) Original query; (b) Follow-up query.



(a) Original query



(b) Follow-up query

Figure 4.2: An example of Google MPSite using Chinese query: the first result of original query is missing after adding site:.au (a) Original query; (b) Follow-up query.

4.1.3 Threats to validity

With regard to the internal validity of this experiment, all the codes were checked carefully and the search engines were set to return all the results they retrieved. According to the support documents for the four search engines [52–55], using the term “site” in the query meant we would get results from a specified site or domain. Therefore, if the original query retrieved a certain result, then the result should appear in the corresponding follow-up query; otherwise, there is a failure in the search engine. In this way, we can use ROCOF to measure the reliability of the search engine. All the advertisements in advertisement sections of search engines were not included in search results. Of course, search engines might put the advertisements in the main section of search results same as normal results, but they should also follow the rules in their user manuals; otherwise, it was reasonable for users to argue their products were not reliable. Only the results of those hours when all the eight scenarios were tested were used to compare the MPSite hourly ROCOF of different scenarios, and therefore the results were selected from exactly the same hours, which significantly eased the effect of the dynamic change of the Web.

4.1.4 Experimental Results

The box-plot result of this experiment are shown in Figure 4.3. A one-way ANOVA was conducted to compare the differences between the eight scenarios and significant differences were found between them at the $p<0.05$ level [$F(7, 3024)=832.889$, $p<0.001$]. Games-Howell post-hoc comparison method is used in this thesis when post-hoc comparison is needed because our test results have unequal sample size. The result of *post-hoc* comparisons using the Games-Howell test is shown in Table 4.2. The metric used in this experiment is MPSite hourly ROCOF. The table shows that the MPSite hourly ROCOF of Google with English queries ($M=0.0259$, $SD=0.0073$) was smaller

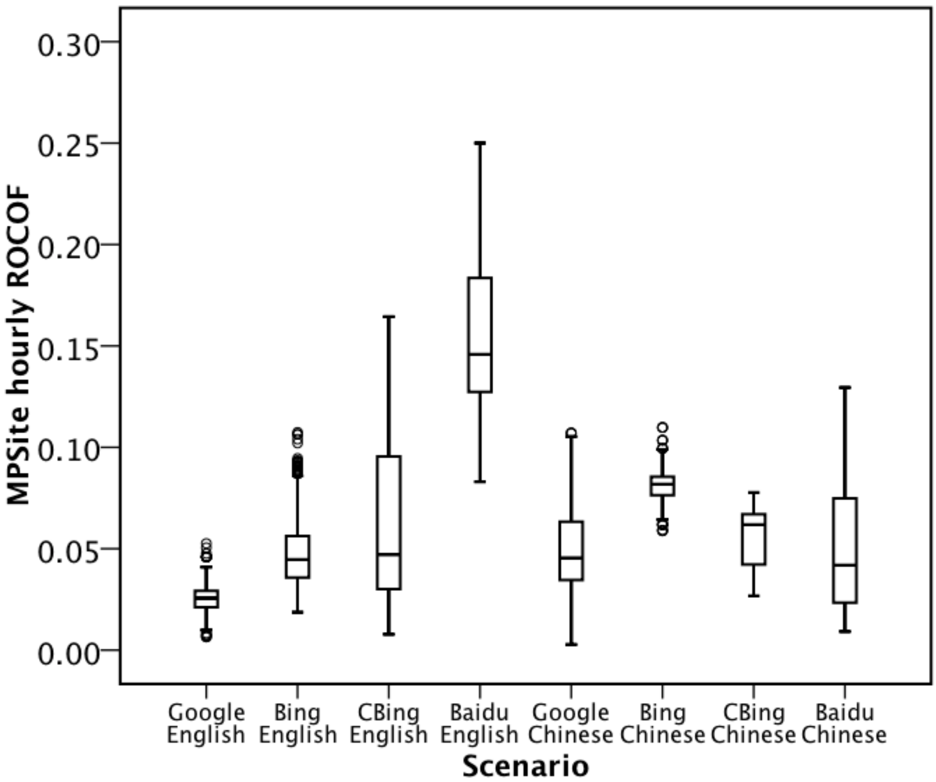


Figure 4.3: MPSite hourly ROCOF of Google, Bing, CBing and Baidu, including English words and Chinese words

than of Google with Chinese queries (M=0.0500, SD=0.0231), and the difference was significant, $p < 0.001$. For Bing, the MPSite hourly ROCOF of English queries (M=0.0491, SD=0.0176) was also significantly smaller than the Chinese (M=0.0809, SD=0.0082), $t(756)=-31.929$, $p < 0.001$, whereas for CBing, the MPSite hourly ROCOF for English queries (M=0.0609, SD=0.0374) was larger than the Chinese queries (M=0.0557, SD=0.0138), $t(756)=2.549$, $p=0.011$. Similarly, the result of Baidu with English queries (M=0.1540, SD=0.0418) was also significantly larger than the Chinese queries (M=0.0523, SD=0.0337), $t(756)=36.908$, $p < 0.001$.

In the English scenario, Google had the smallest MPSite hourly ROCOF and Baidu had the largest MPSite hourly ROCOF, but in the Chinese scenario, there was no significant difference between the MPSite hourly ROCOF of Google and that of Baidu. The MPSites hourly ROCOF of Google and Baidu were significantly smaller than Bing and CBing in Chinese scenario. The MPSite hourly ROCOF of CBing was smaller than Bing in Chinese scenario, while the MPSite hourly ROCOF of CBing was larger than Bing in English scenario which means that CBing was more reliable than Bing in the Chinese scenario and Bing was more reliable than CBing in the English scenario when MPSite hourly ROCOF was used as the metric.

From the results, we can see that search engines may perform different in different language scenarios. For example, MPSite hourly ROCOF of English queries was significantly smaller than the Chinese for Bing. It may be because Bing had more English users than Chinese users, therefore, it was better trained in English language search. Another reason for this may be that Bing was better designed for English query search than Chinese query search. On the contrast, for CBing, the MPSite hourly ROCOF for English queries was larger than the Chinese queries, which may be because CBing was better designed for Chinese language search or because CBing was better trained in Chinese language search. These two reasons are the main reasons why one search engine performs different in different language scenarios.

4.2 MR: MPTitle

4.2.1 Objectives of the Experiment

This experiment is designed to test the search engines' page retrieval capability, focusing on their capability of abstracting a page and understanding user intent.

4.2.2 Experimental Design

4.2.2.1 Independent and Dependent Variables

The independent and dependent variables of this experiment are listed below:

Table 4.2: Multiple comparisons of MPSite hourly ROCOA, using the Games-Howell procedure. The mean differences in highlighted cells are significant at 0.05 level

Multiple Comparisons: MPSite hourly ROCOF			
Games-Howell			
(I) Scenario	(J) Scenario	Mean Difference (I-J)	Sig.
Within Single Search Engine			
Google English	Google Chinese	-0.0241	<0.001
Bing English	Bing Chinese	-0.0319	<0.001
CBing English	CBing Chinese	0.0052	0.178
Baidu English	Baidu Chinese	0.1018	<0.001
Between Search Engines			
Google English	Bing English	-0.0232	<0.001
Google English	CBing English	-0.0350	<0.001
Google English	Baidu English	-0.1282	<0.001
Bing English	CBing English	-0.0119	<0.001
Bing English	Baidu English	-0.1050	<0.001
CBing English	Baidu English	-0.0931	<0.001
Google Chinese	Bing Chinese	-0.0309	<0.001
Google Chinese	CBing Chinese	-0.0057	0.001
Google Chinese	Baidu Chinese	-0.0022	0.964
Bing Chinese	CBing Chinese	0.0252	<0.001
Bing Chinese	Baidu Chinese	0.0287	<0.001
CBing Chinese	Baidu Chinese	0.0034	0.597

Independent variables: language (English or Chinese), search engines (Google, Bing, CBing or Baidu)

Dependent Variable: MPTitle hourly ROCOA.

According to the independent variables there are eight scenarios: Google English, Bing English, CBing English, Baidu English, Google Chinese, Bing Chinese, CBing Chinese, and Baidu Chinese. The experiment tested the MPTitle of each of the eight scenarios and then compared their MPTitle hourly ROCOA.

4.2.2.2 Experimental Procedures

The experiment procedures were the same as for the previously mentioned MPSite, apart from how the follow-up queries were generated. Instead of adding the top level domain name of each result of the original query, this section added the title of each result of the original query to the original query. Before the title of each result of the original query was added to obtain a follow-up query, all the punctuations were removed from the title.

The original query and follow-up query of this experiment were defined as below:

Original query: Randomly select a query *"A"* (with quotes) with fewer than 20 results. The way to come up with a query is the same as in Section 4.1.

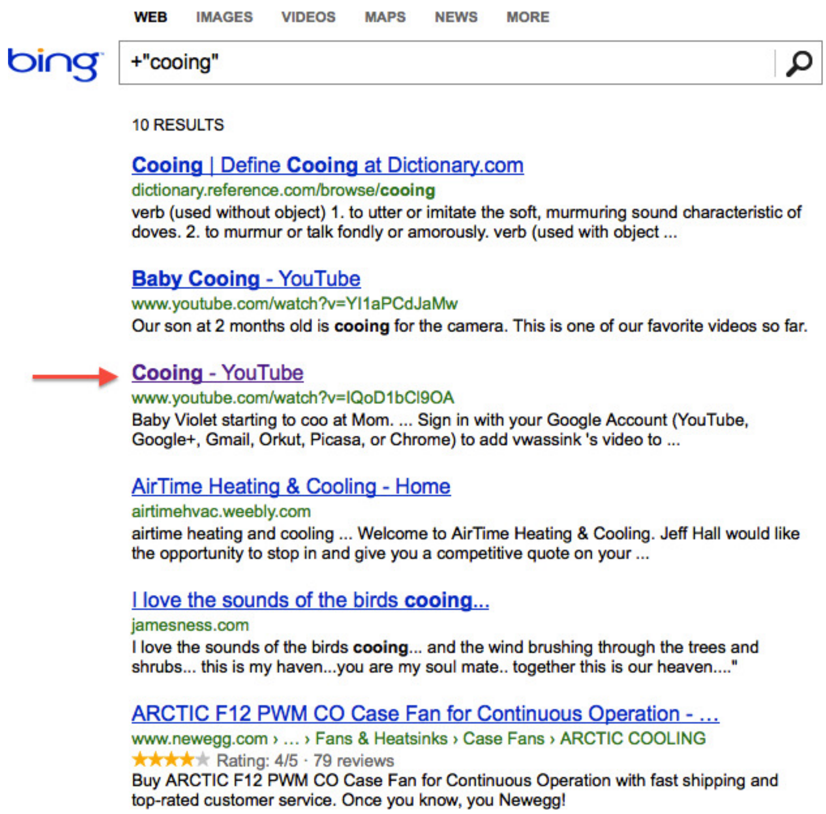
Follow-up queries: *"A"* (with quotes) + [the title of each result of the original query], the i^{th} follow-up query is the one added the title of the i^{th} result of the original query.

Figure 4.4 is an example of the missing page of Bing with the original query '+"cooing"'. According to the help page of Bing [56], we can find webpages that contain all the terms that are preceded by the "+" symbol, where the "+" symbol allows for the inclusion of terms that are usually ignored. In Bing and CBing, query term *"A"* was preceded by the "+" symbol, but no "+" symbol was applied to the words in title because the title is a description generated by the search engine rather than a strong copied from the target Web page. The figure shows that the title of the third result of the original query was “Cooing - YouTube”. The title of a result should be closely related to the result, which means the title should either contain some phrases of the results or briefly summarise the result page. We removed the punctuations from the title and added it to the original query to get ‘+“cooing” Cooing YouTube’ as the follow-up query. Obviously, the follow-up query only takes keywords from the title of the third result and it should be able to retrieve this result. However, the third result of the original query was missing after adding title.

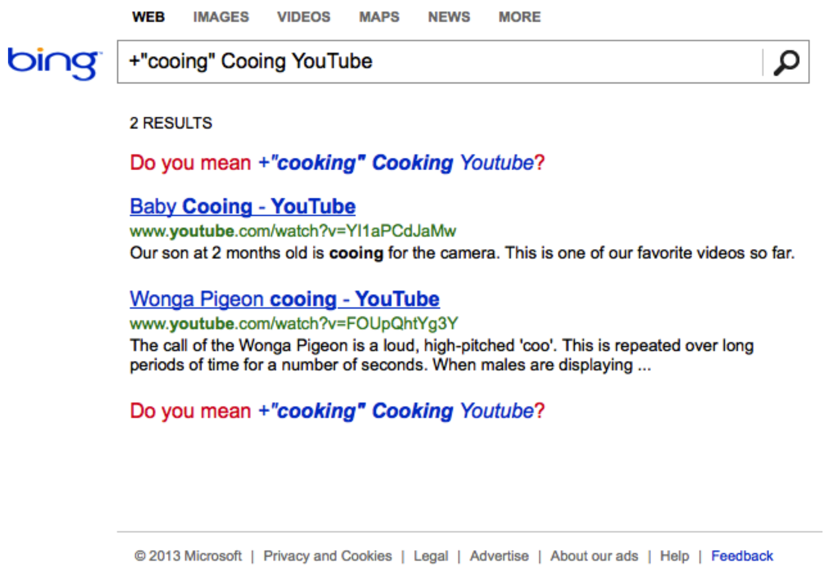
This experiment was conducted to evaluate the MPTitle hourly ROCOA of different scenarios. Table 4.3 shows the numbers of test case pairs which were used to compare the MPTitle hourly ROCOA of different scenarios. The table only shows the number of test case pairs of those hours when all eight scenarios were tested. According to the table 380 hours of data were used to compare the MPTitle hourly ROCOA of the eight different scenarios and the total number of test case pairs was about 7,600,000. Because all the test cases used in every hour were randomly selected, it is infeasible to include the search set in this thesis.

Table 4.3: The number of test case pairs used to compare MPTitle hourly ROCOA

Search Engine	Usage Pattern	Test case pairs per hour (approximate)	Hours	Total test case pairs (approximate)
Google	English	1000	380	380,000
	Chinese	1000	380	380,000
Bing	English	3000	380	1,140,000
	Chinese	3000	380	1,140,000
CBing	English	3000	380	1,140,000
	Chinese	3000	380	1,140,000
Baidu	English	3000	380	1,140,000
	Chinese	3000	380	1,140,000



(a) Original query



(b) Follow-up query

Figure 4.4: An example of Bing MPTitle: the third result of original query is missing after adding title (a) Original query; (b) Follow-up query.

4.2.3 Threats to Validity

The main concern in the experiment with validity is the correctness of the MR MPTitle. The title of a result should be closely related to the result, which means the title should either contain some phrases of the results or briefly summarise the result page. Therefore, in this experiment, if the title of a result is added to the original query, the follow-up query should also be able to retrieve this result; otherwise the user can reasonably assume there is an anomaly. This anomaly may either come from the bad title presented by the search engine or from problems in the retrieval algorithm. Only the results of those hours when all eight scenarios were tested were used to compare MPTitle hourly ROCOA of different scenarios, and therefore the results were selected from exactly the same hours, which significantly eased the effect of the dynamic change of the Web.

4.2.4 Experimental Results

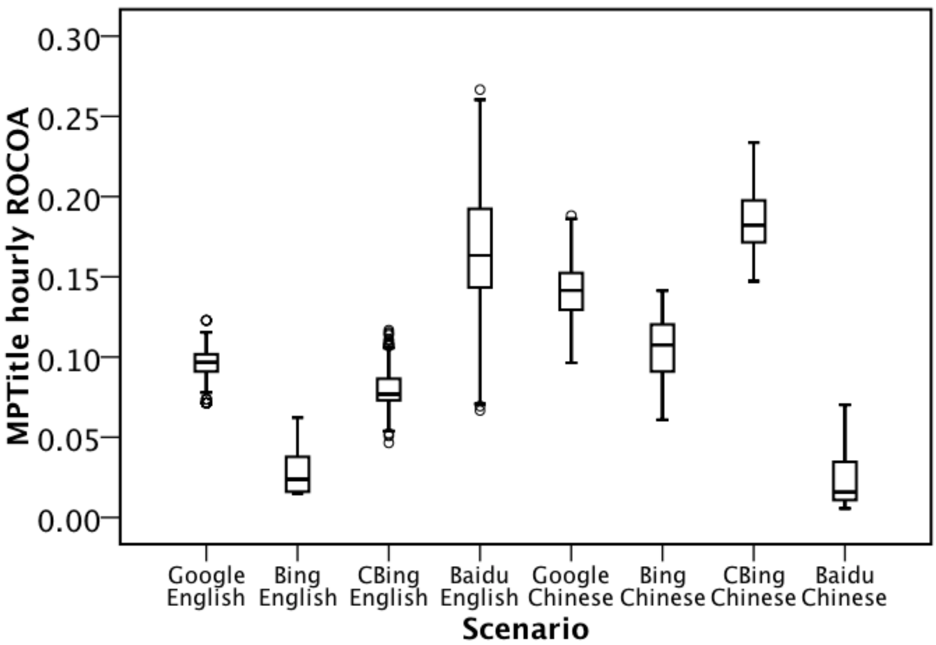


Figure 4.5: MPTitle hourly ROCOA of Google, Bing, CBing and Baidu, including English words and Chinese words

Table 4.4: Multiple comparisons of MPTitle hourly ROCOA, using the Games-Howell procedure. The mean differences in highlighted cells are significant at 0.05 level

Multiple Comparisons: MPTitle hourly ROCOA			
Games-Howell			
(I) Scenario	(J) Scenario	Mean Difference (I-J)	Sig.
Within Single Search Engine			
Google English	Google Chinese	-0.0462	<0.001
Bing English	Bing Chinese	-0.0773	<0.001
CBing English	CBing Chinese	-0.1044	<0.001
Baidu English	Baidu Chinese	0.1430	<0.001
Between Search Engines			
Google English	Bing English	0.0679	<0.001
Google English	CBing English	0.0153	<0.001
Google English	Baidu English	-0.0701	<0.001
Bing English	CBing English	-0.0526	<0.001
Bing English	Baidu English	-0.1380	<0.001
CBing English	Baidu English	-0.0854	<0.001
Google Chinese	Bing Chinese	0.0367	<0.001
Google Chinese	CBing Chinese	-0.0429	<0.001
Google Chinese	Baidu Chinese	0.1190	<0.001
Bing Chinese	CBing Chinese	-0.0796	<0.001
Bing Chinese	Baidu Chinese	0.0823	<0.001
CBing Chinese	Baidu Chinese	0.1619	<0.001

The experimental results are shown in Figure 4.5. The page retrieval capability of each search engine differed between the English queries and Chinese queries. A one-way ANOVA was conducted to compare the differences between the eight scenarios on MPTitle hourly ROCOA and significant differences were found at the $p<0.05$ level [$F(7, 3032)=3505.519$, $p<0.001$]. Table 4.4 shows the result of post-hoc comparisons using the Games-Howell test. Comparisons within single search engine shows that the missing page rate for Google with English queries (M=0.0953, SD=0.0108) was smaller than Google with Chinese queries (M=0.1415, SD=0.0180), and the difference was significant, $t(758) = -42.938$, $p <0.001$. For Bing, the missing page rate of English queries (M=0.0274, SD=0.0114) was also significantly smaller than the Chinese queries (M=0.1048, SD=0.0174), $t(758)=-72.390$, $p<0.001$. Similarly, for CBing, the missing page rate for English queries (M=0.0800, SD=0.0122) was significantly smaller than that of Chinese queries (M=0.1844, SD=0.0187), $t(758)=-91.071$, $p<0.001$. However, the result of Baidu with English queries (M=0.1654, SD=0.0388) was significantly larger than the Chinese queries (M=0.0225, SD=0.0140), $t(758)=67.616$, $p <0.001$. The two reasons discussed in section 4.1.4 can also be used to explain the result in this experiment.

The table shows that in the English language scenario, the MPTitle hourly ROCOA of Bing is significantly smaller than Google, CBing and Baidu while the MPTitle hourly ROCOA of Baidu is significantly larger than the others. In the Chinese scenario, the MPTitle hourly ROCOA of any two of the four search engines are also significantly different. Of the four search engines, Baidu had the smallest MPTitle hourly ROCOA and CBing had the largest MPTitle hourly ROCOA. This means that Baidu had the best quality among the four search engines in the Chinese scenario when the MPTitle hourly ROCOA was used as the metric.

For each search engine under test, it is important for developers and users to know its strength and weakness. The experimental results of MPSite and MPHeading show that: 1. metamorphic testing can provide an answer to this question in terms of the search engines' performance under different operational profiles; 2. weakness or faults are unevenly distributed across the search engines' features (in other words, the qualities of different features of the same search engine are not equal); for example, Google English performed best in its "site" feature but only the fourth in the "heading" feature, among the eight scenarios. This explains why some scenarios had very different performance when tested against different MRs. As a result, the recommendation is that more than one MR should be used in testing in order to cover different features of the search engines.

4.3 MR: MPReverseJD

4.3.1 Objectives of the Experiment

The aim is to test the search engines' page retrieval capability, focusing on their insensitivity to similar queries that only differ in word order.

4.3.2 Experimental Design

4.3.2.1 Independent and Dependent Variables

The independent and dependent variables of this experiment are listed below:

Independent variables: word categories (Person names, Company names or drug names), search engines (Google, Bing, CBing or Baidu).

Dependent variable: Hourly average SRJC, which is defined in Equation 3.1.

4.3.2.2 Experimental Procedures

Names were randomly selected from each name category and combined as a query which was then used to query search engines. At first, two names was selected from one name category and sent to search engines. If the result count of this query was smaller than 20 then it could be used as the original query. Otherwise, another name selected from the same name category would be added to the query and sent to search engine. If the result count was still larger than 20 then the fourth name would be added to the query. At most four names were used in each query, and the names in one query were from the same name category. The follow-up query consisted of the names in original query but in reverse order. In this experiment, quotation marks were used to bracket every single names.

The original query and follow-up query in the experiment were defined as:

Original query: "A_1" + "A_2" [+ "A_3"] [+ "A_4"] (The names inside the brackets were optional, therefore the query may include 2 to 4 names. If "A_1"+ "A_2" or "A_1"+ "A_2" + "A_3" had fewer than 20 results, then the remaining names are not needed.)

Follow-up query: ["A_4" +]["A_3" +] "A_2" + "A_1"

In the original query and follow-up query, $A_i\,(i \in N,\, 0 < i < 5)$ were randomly selected from one of the three name categories below:

Category 1: 200 person names.

Category 2: 200 company names.

Category 3: 200 drug names.

The names are include in Appendix A and all names were in English. Figure 4.6 is an example of the MPReverseJD of Baidu with the original query " 'Becampicillin' 'Aspirin' 'Flecainide' ". The figure shows that, the result count of the original query was equal to two which is less than 20. The order of the three drug names were reversed to get the follow-up query " 'Flecainide' 'Aspirin' 'Becampicillin' " which retrieved 28 results. We compared the 28 results with the two results of original query and found that the two results of the original query were included in the 28 results of the follow-up query. In this example, $|\{original\_query\_results\} \cap \{follow\_up\_query\_results\}|$ $= 2$ and $|\{original\_query\_results\} \cup \{follow\_up\_query\_results\}| = 28$, therefore, the metric SRJC is 0.0714.

Table 4.5 shows the number of test case pairs which were used to compare the SRJC of each search engine on different word categories. To study the effect of different categories of words on the SRJC, we compared the differences between the hourly average SRJC of different word categories of the same search engine and to minimise the effect of testing time on the result we only used the results of those hours when all the three word categories had been tested.

The hourly average SRJC of the three word categories for Google, Bing, CBing and Baidu were calculated from data of 150 hours, 452 hours, 205 hours, and 185 hours, respectively and the number of test case pairs tested for the four search engines were approximately 225,000, 1,356,000, 615,000 and 555,000, respectively. In total, about 2,751,000 test case pairs were tested in this experiment.

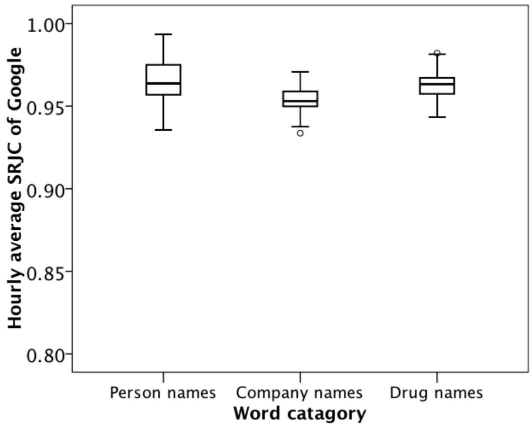
Table 4.5: The number of test case pairs in the experiment MPReverseJD

Search Engine	Usage Pattern	Test case pairs per hour (approximate)	Hours	Total test case pairs (approximate)
Google	Person	500	150	75,000
	Company	500	150	75,000
	Drug	500	150	75,000
Bing	Person	1000	452	452,000
	Company	1000	452	452,000
	Drug	1000	452	452,000
CBing	Person	1000	205	205,000
	Company	1000	205	205,000
	Drug	1000	205	205,000
Baidu	Person	1000	185	185,000
	Company	1000	185	185,000
	Drug	1000	185	185,000

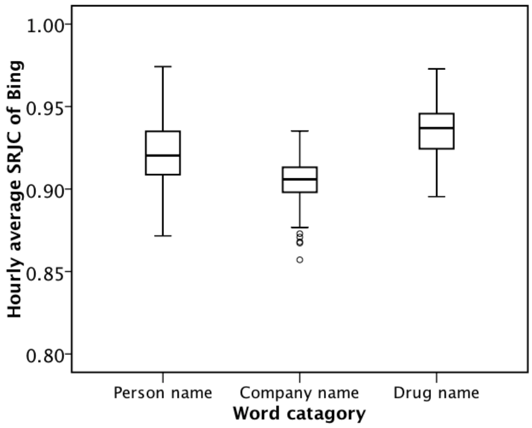
(a) Original query: retrieved two results

(b) Follow-up query: retrieved 28 results

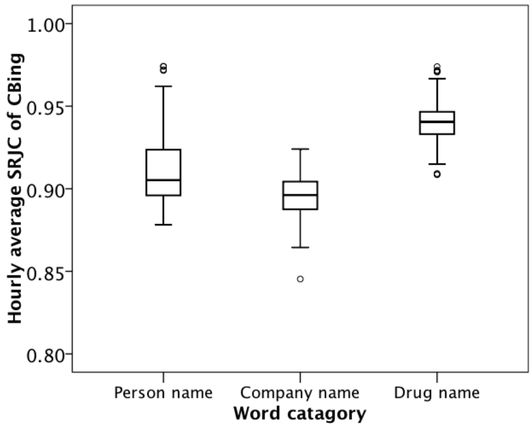
Figure 4.6: An example of Baidu MPReverseJD (a) Original query; (b) Follow-up query.



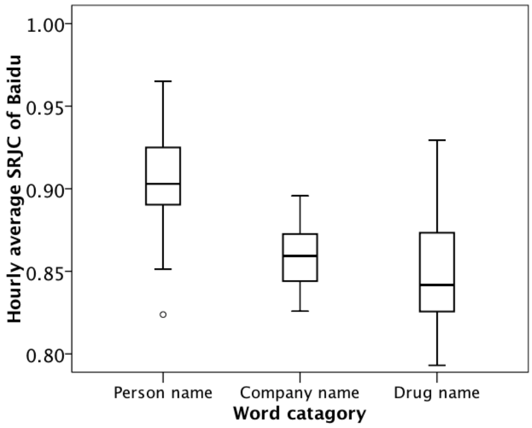
(a)Google



(b)Bing



(c)CBing



(d)Baidu

Figure 4.7: Hourly average SRJC of search engines on three different word categories.

4.3.3 Threats to Validity

The main concern in the experiment with validity was the correctness of the MR MPReverseJD. $A_i\,(i \in N,\, 0 < i < 5)$ in a query were selected from only one word category and are all names, so that the semantic of the reversed order query was similar to the original query and the keywords in the two queries were the same. On this basis it was reasonable to believe they would return a large number of common results, if the search engine were in good quality.

Table 4.6: Multiple comparisons of hourly average SRJC, using the Games-Howell procedure. The mean differences in highlighted cells are significant at 0.05 level

Multiple Comparisons: MPReverseJD hourly average SRJC			
Games-Howell			
(I) Scenario	(J) Scenario	Mean Difference (I-J)	Sig.
Within Single Search Engine			
Google Person Name	Google Company Name	0.0114	<0.001
Google Person Name	Google Drug Name	0.0027	0.414
Google Company Name	Google Drug Name	-0.0087	<0.001
Bing Person Name	Bing Company Name	0.0174	<0.001
Bing Person Name	Bing Drug Name	-0.0122	<0.001
Bing Company Name	Bing Drug Name	-0.0296	<0.001
CBing Person Name	CBing Company Name	0.0168	<0.001
CBing Person Name	CBing Drug Name	-0.0289	<0.001
CBing Company Name	CBing Drug Name	-0.0458	<0.001
Baidu Person Name	Baidu Company Name	0.0472	<0.001
Baidu Person Name	Baidu Drug Name	0.0576	<0.001
Baidu Company Name	Baidu Drug Name	0.0105	0.004
Between Search Engines			
Google Person Name	Bing Person Name	0.0416	<0.001
Google Person Name	CBing Person Name	0.0523	<0.001
Google Person Name	Baidu Person Name	0.0583	<0.001
Google Company Name	Bing Company Name	0.0475	<0.001
Google Company Name	CBing Company Name	0.0577	<0.001
Google Company Name	Baidu Company Name	0.0941	<0.001
Google Drug Name	Bing Drug Name	0.0266	<0.001
Google Drug Name	CBing Drug Name	0.0206	<0.001
Google Drug Name	Baidu Drug Name	0.1132	<0.001
Bing Person Name	CBing Person Name	0.0107	<0.001
Bing Person Name	Baidu Person Name	0.0167	<0.001
Bing Company Name	CBing Company Name	0.0102	<0.001
Bing Company Name	Baidu Company Name	0.0465	<0.001
Bing Drug Name	CBing Drug Name	-0.0060	<0.001
Bing Drug Name	Baidu Drug Name	0.0866	<0.001
CBing Person Name	Baidu Person Name	0.0060	0.319
CBing Company Name	Baidu Company Name	0.0364	<0.001
CBing Drug Name	Baidu Drug Name	0.0926	<0.001

4.3.4 Experimental Results

The box-plot result of hourly average SRJC for each search engine is presented in Figure 4.7. The vertical axis of any individual subfigure is the hourly average SRJC. A one-way ANOVA was conducted to compare the differences between the hourly average SRJC of each search engine on different word categories and the results shows that there were significant differences between different scenarios. Table 4.3.4 shows the result of post-hoc comparisons using the Games-Howell test of each search engine. SRJC is the similarity between the two query result sets and therefore the bigger the value, the less sensitive the search engine is.

It can be seen from the figures that Google was the least sensitive search engines in terms of the page retrieval capability. Google, Bing, and CBing had similar patterns of page retrieval capability in these three word categories in that they all performed best on drug names and worst on company names, but Baidu performed the best on person names and worst on drug names. Of these four search engines Google obtained the biggest hourly average SRJC value on all three word categories, while Baidu obtained the smallest hourly average SRJC value on all three word categories. Comparisons between search engines show that Google had the largest hourly average SRJC values on the three word categories while Baidu had the smallest value.

Chapter 5

Empirical Evaluation Using the MRs of Swapping Keywords

5.1 MR: Universal SwapJD

5.1.1 Objective of the Experiment

The goal of this experiment is to test the search engines' consistency in page ranking, focusing on their insensitivity to similar queries that only differ in word order.

5.1.2 Experimental Design

5.1.2.1 Independent and Dependent Variables

Independent variable: Search engines (Google, Baidu, Bing and CBing).

Dependent variable: Hourly average JCT50, which is defined in Equation 3.2.

In this experiment all search engines use the same queries in each hour. JCT50 is the Jaccard coefficient of the top 50 results of the original query results and the top 50 results of the follow-up query results. Thus, this experiment only consider the top 50 results of the original queries and follow-up queries.

5.1.2.2 Experimental Procedures

This experiment selected two words from two of the three pre-designed word lists to obtain the original query and swapped the two words in the original query to obtain the follow-up query. This experiment recorded the OQ50 and the FQ50 and then calculate the overlap between them using the formula define in section 5.1.2.1.

The pre-designed query list is defined as below:

List one: London, Stockholm, Berlin, Antwerp, Paris, Amsterdam, Tokyo, Helsinki, Sydney, Rome, Montreal, Moscow, Seoul, Barcelona, Atlanta, Athens, Beijing, Toronto, Oslo and Melbourne (20 city names)

List two: morning, afternoon, evening, midnight, today, tomorrow, yesterday (7 words)

List three: movie, song, music, book, game, story, magazine, food, shop, car, weather, olympics, library, school, airport, bus, newspaper, traffic, population, pollution (20 words)

The reason why this experiment could test the search engines' consistency in page ranking was that this experiment focus on the top 50 results of queries, if search engines ranked the top 50 results of the original query out of the top 50 in the follow up query, then the score of the SwapJD would be very low.

The original query and follow up query of this experiment were defined as:

Original query: $A + B$, where A and B were selected from different word lists defined above.

Follow-up query: $B + A$, which was obtained by swapping the keywords of original query.

In order to demonstrate the example easily we chose a special example with a small result count. Figure 5.1 is an example of the swapping keywords of Bing on the 13 January 2014 with the original query 'Seoul traffic'. The figure shows that the result count of the original query was 25 while we swapped the two keywords to obtain the follow-up query 'traffic Seoul' which did not retrieve any result. In this example, $JCT50{=}|\{OQ50\} \cap \{FQ50\}| = 0$, therefore the hourly average JCT50 was zero. All examples in this thesis were repeatable in the time when it was repeatable at the time of experiment.

The three lists contain 20 words, 7 words and 20 words, respectively so the total number of two word combinations is 20*7+ 20*20+7*20=680. In this experiment these 680 original queries and the 680 follow-up queries were queried every hour. This experiment only focused on the top 50 results of the queries, even though the search engines returned a large number of results, because most people are not be interested in the results beyond the top 50. Table 5.1 shows the number of test case pairs tested in this experiment.

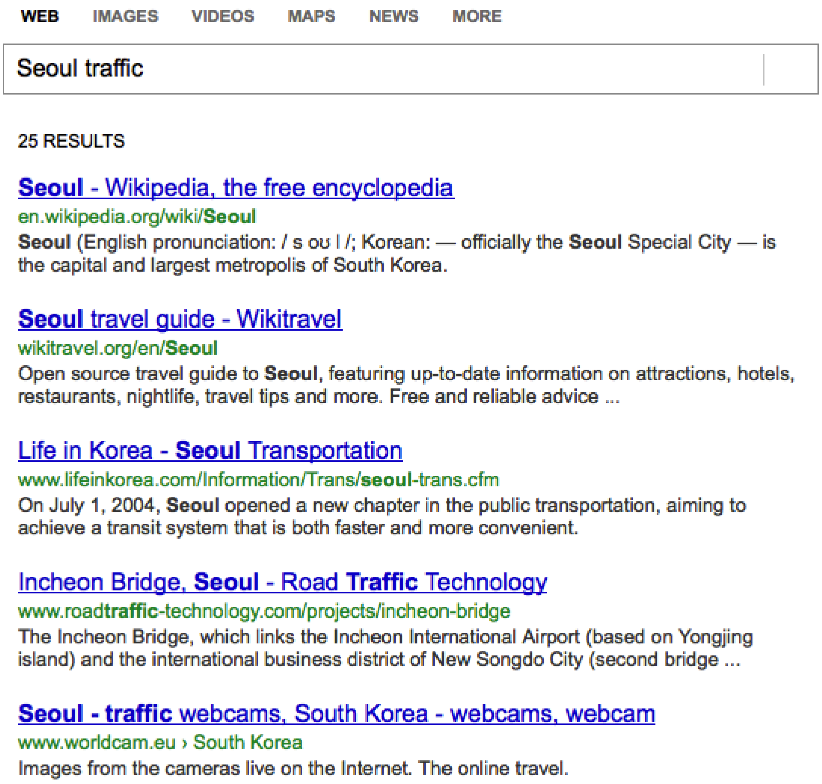
Table 5.1: The number of test case pairs in the experiment universal SwapJD

Search Engine	Usage Pattern	Test case pairs per hour	Hours	Total test case pairs
Google	universal	680	548	372,640
Baidu	universal	680	548	372,640
Bing	universal	680	548	372,640
CBing	universal	680	548	372,640

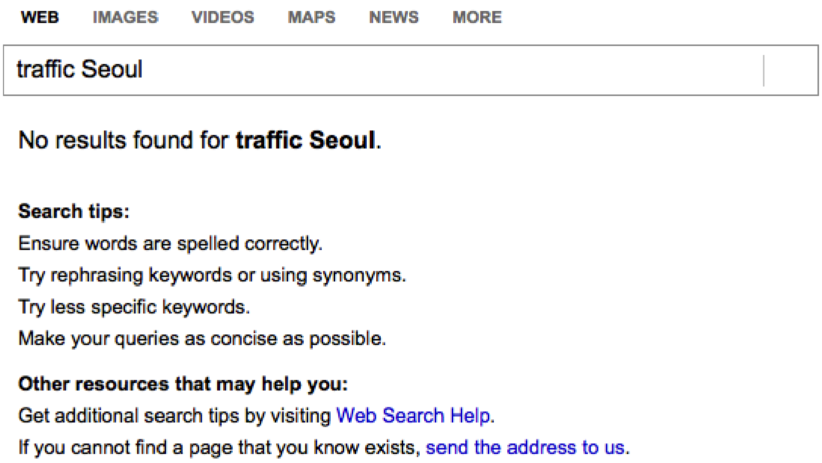
5.1.3 Threats to Validity

The main concern in the experiment with validity was the correctness of the MR SwapJD. Both of the two words in one query were nouns, so the semantic of the reversed order query was similar to the original query in most cases and the keywords in the two queries were the same. On this basis it was reasonable to believe they would return a large number of common results in their top 50 results, if the search engine were in good quality.

We sent the same original query and the same follow-up query as the ones shown



(a)Original query: retrieved 25 results



(b)Follow-up query: did not retrieve any result

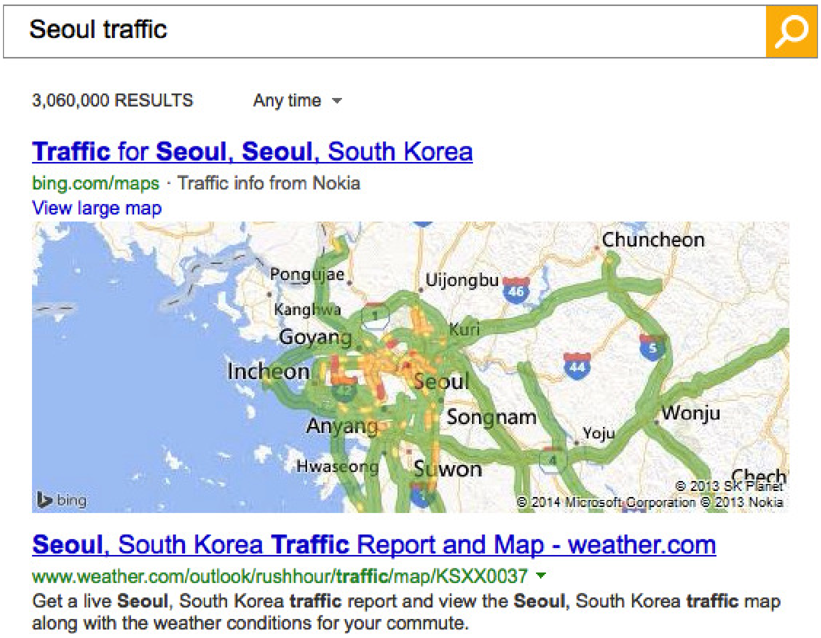
Figure 5.1: An example of Bing SwapJD on 13 January 2014 (a) Original query; (b) Follow-up query.

in Figure 5.2 to Bing on 27 February 2014. The original query 'Seoul traffic' retrieved 3,060,000 results and the follow-up query retrieved 3,000,000 results and the OQ50 and the FQ50 had 36 common results so the JCT50 was equal to 0.5625. That is to say, the original query result and the follow-up query in this experiment could retrieve a large amount of common results in the top 50 results. However, the search engines sometimes do not perform as their designers expected and this was one of the motivations for this research.

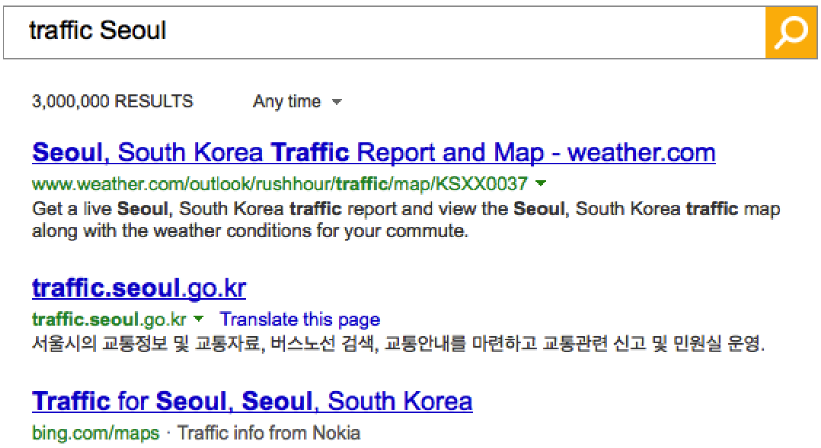
5.1.4 Experimental Results

Figure 5.1.4 shows the hourly average JCT50 of different search engines. The figure only shows the results of those hours when all four search engines were tested, and of these four search engines, Google had the highest SwapJD value of 0.9138. This means the common rate of the top 50 results of original query and the follow-up query was the largest. Meanwhile, Baidu, Bing and CBing had smaller SwapJD scores, with values of 0.5299, 0.5175 and 0.5422, respectively.

A one-way ANOVA was conducted to compare the differences between search engines on hourly average JCT50 and the result showed that there were significant differences between Google, Bing, CBing and Baidu on hourly average JCT50 at the $p<0.05$ level [$F(3, 2188)=21553.160$, $p<0.001$]. Table 5.1.4 shows the result of post-hoc comparisons using the Games-Howell test where the hourly average JCT50 of any two of the four search engines were significantly different. The hourly average JCT50 of Google was significantly larger than Baidu, Bing and CBing. Also, the hourly average JCT50 of CBing was significantly larger than Baidu and Bing while hourly average JCT50 of Bing was significantly smaller than that of Baidu, which means that Google was the most consistent of the four search endings tested and Bing was the least consistent when hourly average JCT50 was used as the metric. In other words, Google



(a)Original query



(b)Follow-up query

Figure 5.2: An example of Bing SwapJD on 27 February 2014 (a) Original query; (b) Follow-up query.

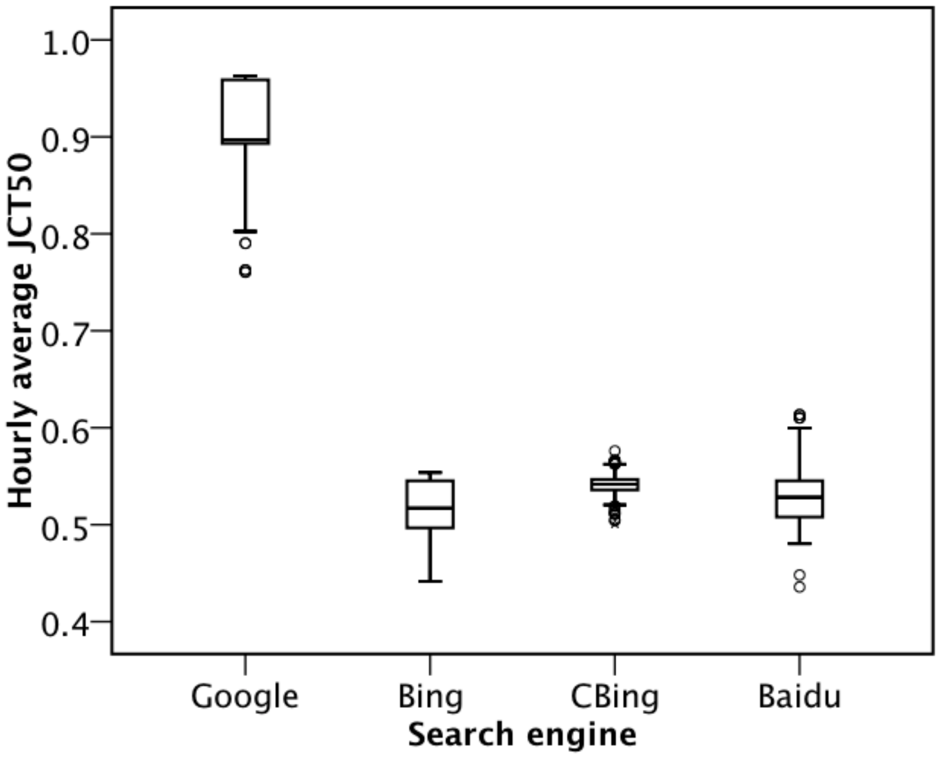


Figure 5.3: Hourly average JCT50 of Google, Bing, CBing and Baidu

had the highest proportion of the overlap between original query result set and follow up query result set. That is because Google was more insensitive to similar queries that only differ in word order in terms of the page ranking function.

In this SwapJD experiment, the original query and the follow up query had similar meaning in most cases, so a possible reason for why Google performed the best amongst the four search engines is that Google had the best ability in semantic search.

We can look back at the section of MPTitle, Google performed the third in both English language scenario and Chinese language scenario. Because MPTitle used the result title to search, the performance of search engines on MPTitle is affected by the following two abilities, namely the ability of generating proper title of the search results and the ability of synonym based search. However in Universal SwapJD section, Google had the largest SwapJD value, which means the synonym based search ability of Google was good. Therefore, we may get a conclusion that the reason why Google

Table 5.2: Multiple comparisons of hourly average JCT50 in MR Universal SwapJD, using the Games-Howell procedure. The mean differences in highlighted cells are significant at 0.05 level

Multiple Comparisons: Universal SwapJD hourly average JCT50			
Games-Howell			
(I) Scenario	(J) Scenario	Mean Difference (I-J)	Sig.
Google Universal	Bing Universal	0.3962	<0.001
Google Universal	CBing Univeral	0.3716	<0.001
Google Universal	Baidu Universal	0.3939	<0.001
Bing Universal	CBing Univeral	-0.0246	<0.001
Bing Universal	Baidu Universal	-0.0023	0.998
CBing Univeral	Baidu Universal	0.0223	<0.001

did not perform the best in MPTitle is that the ability in generating proper title of Google was not the best among the four search engines.

5.2 MR: Swapping Keywords with Domain

5.2.1 Objective of the Experiment

The previous section discussed the Universal SwapJD of the four search engines. There is a research question which needs to be discussed: Will the domain scale affect the SwapJD value? The purpose of this experiment is to address the question.

5.2.2 Experimental Design

5.2.2.1 Independent and Dependent Variables

Independent variables: Search engines (Google, Baidu, Bing or CBing), domain name (site:.com, site:.edu, site:.mil or site:.lc)

Dependent variable: Hourly average JCT50, which is defined in Equation 3.2.

In this experiment all the search engines used the same query words.

5.2.2.2 Experimental Procedures

A domain name was added to the original queries and the follow-up query described in Section 5.1. For example, an original query was "London morning site:.com" (without double quotes) and the corresponding follow-up query was "morning London site:.com" (without double quotes). In this way the original query results and the follow-up query results were in same domain. Same as Section 5.1, this experiment recorded the OQ50 and the FQ50 and then calculate the overlap between them using the Equation 3.2 to get the hourly average JCT50 value.

The original query and follow up query of this experiment were defined as:

Original query: A + B + site:[one of these four domain name: ".com", ".edu", ".mil" and ".lc"]

Follow-up query: B + A + site:[the same domain name as the original query]

In the original query and follow-up query, A and B were selected from two of the three word lists defined in Section 5.1. The follow-up query used the same domain name as the original query, so that the only difference between the two queries was the order of A and B.

Table 5.3: The number of test case pairs in the experiment SwapJD with domain

Search Engine	Usage Pattern	Test case pairs per hour	Hours	Total test case pairs
Google	site:.com	680	103	70,040
	site:.edu	680	103	70,040
	site:.mil	680	103	70,040
	site:.lc	680	103	70,040
Baidu	site:.com	680	100	68,000
	site:.edu	680	100	68,000
	site:.mil	680	100	68,000
	site:.lc	680	100	68,000
Bing	site:.com	680	148	100,640
	site:.edu	680	148	100,640
	site:.mil	680	148	100,640
	site:.lc	680	148	100,640
CBing	site:.com	680	131	89,080
	site:.edu	680	131	89,080
	site:.mil	680	131	89,080
	site:.lc	680	131	89,080

Table 5.3 shows the number of test case pairs tested in this experiment. Table 5.4 shows the average result counts of different scenarios. The average result counts are calculated as the average of the 680 original queries and 680 follow-up queries. In this table, the result counts of Universal SwapJD are also included. It is obviously that in all the four search engines, the average result counts of the four domains has the following property: ".com" > ".edu" > ".mil" > ".lc".

Table 5.4: Average number of result counts

	site:.com	site:.edu	site:.mil	site:.lc
Google	77545421.32	2144545.06	27893.25	5952.32
Bing	11597638.90	553849.31	42809.25	39.74
CBing	8383810.22	247197.85	31920.81	29.16
Baidu	2553871.82	81266.46	8.59	2.73

5.2.3 Threats to Validity

The threats to the validity of this experiment were the same as the experiment of the MR Universal SwapJD.

5.2.4 Experimental Results

The box-plot results of hourly average JCT50 of the four search engines are shown in Figure 5.4. Each sub-figure only shows the results of each search engine of those hours when all four domain names were tested. One-way ANOVA result shows there are significant differences among these scenarios. Table 5.2.4 shows the result of post-hoc comparisons using the Games-Howell test.

It can be seen from the figures that the four search engines had the same pattern on SwapJD in the four domains; they all had the highest hourly average JCT50 in "site:.lc", and the smallest value in "site:.com". The hourly average JCT50 of "site:.edu" was

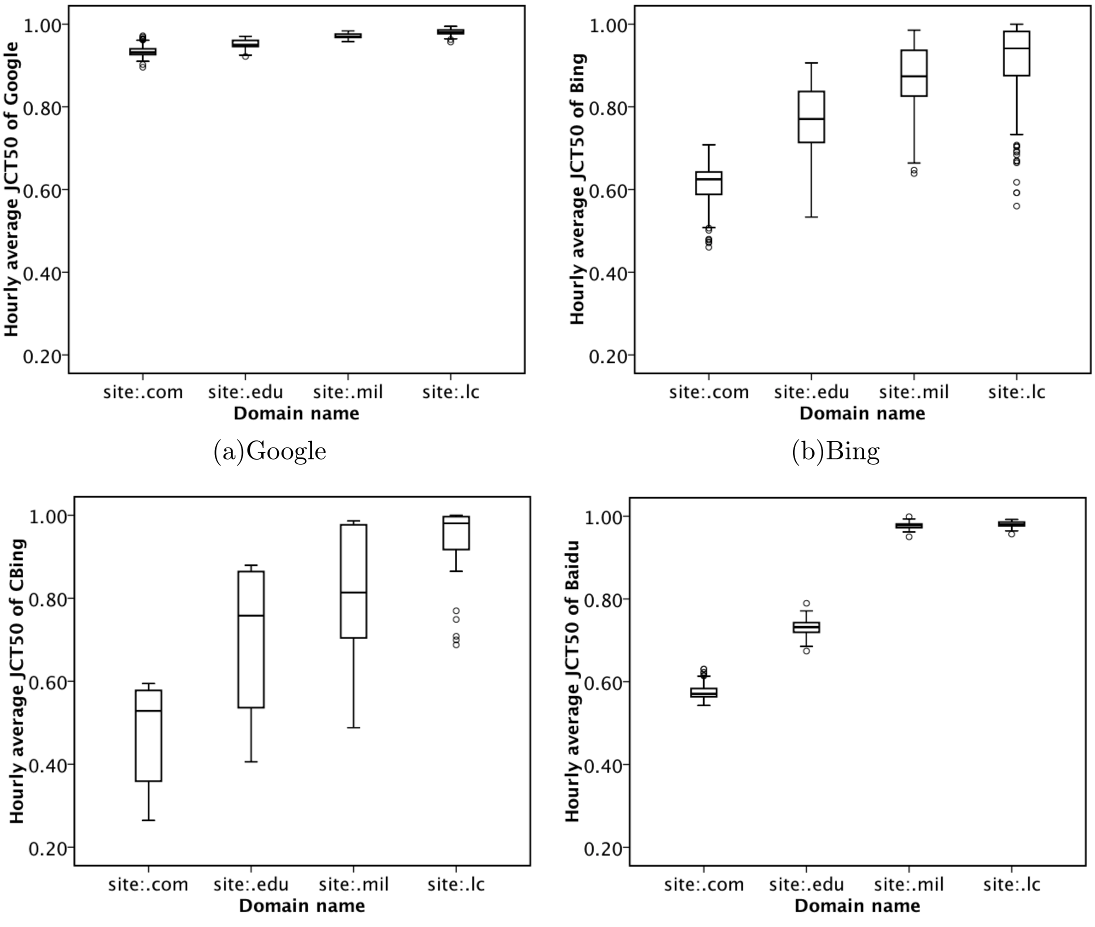


Figure 5.4: Hourly average JCT50 of search engines in MR SwapJD with Domain

Table 5.5: Multiple comparisons of hourly average JCT50 in SwapJD with Domain, using the Games-Howell procedure. The mean differences in highlighted cells are significant at 0.05 level

Multiple Comparisons: SwapJD hourly average JCT50			
Games-Howell			
(I) Scenario	(J) Scenario	Mean Difference (I-J)	Sig.
Within Single Search Engine			
Google Site:.com	Google site:.edu	-0.0166	<0.001
Google Site:.com	Google site:.mil	-0.0371	<0.001
Google Site:.com	Google site:.lc	-0.0461	<0.001
Google site:.edu	Google site:.mil	-0.0205	<0.001
Google site:.edu	Google site:.lc	-0.0295	<0.001
Google site:.mil	Google site:.lc	-0.0090	<0.001
Bing site:.com	Bing site:.edu	-0.1524	<0.001
Bing site:.com	Bing site:.mil	-0.2541	<0.001
Bing site:.com	Bing site:.lc	-0.2951	<0.001
Bing site:.edu	Bing site:.mil	-0.1017	<0.001
Bing site:.edu	Bing site:.lc	-0.1428	<0.001
Bing site:.mil	Bing site:.lc	-0.0411	0.015
CBing site:.com	CBing site:.edu	-0.2283	<0.001
CBing site:.com	CBing site:.mil	-0.3464	<0.001
CBing site:.com	CBing site:.lc	-0.4698	<0.001
CBing site:.edu	CBing site:.mil	-0.1181	<0.001
CBing site:.edu	CBing site:.lc	-0.2415	<0.001
CBing site:.mil	CBing site:.lc	-0.1234	<0.001
Baidu site:.com	Baidu site:.edu	-0.1541	<0.001
Baidu site:.com	Baidu site:.mil	-0.4016	<0.001
Baidu site:.com	Baidu site:.lc	-0.4048	<0.001
Baidu site:.edu	Baidu site:.mil	-0.2475	<0.001
Baidu site:.edu	Baidu site:.lc	-0.2507	<0.001
Baidu site:.mil	Baidu site:.lc	-0.0032	0.215

smaller than “site:.mil” in all four search engines. Most of the differences are significant except the results of “site:.mil” and “site:.lc” of Baidu.

The results show that search engines perform better on smaller scale domain in regarding to hourly average JCT50, which answers the research question posed in this section.

Chapter 6

Empirical Evaluation Using the MRs of No Ranking Drop with Domain

6.1 Objective of the Experiment

The purpose of these experiments is to test the search engines' consistency in page ranking, focusing on their consistency with different domains using the MRs Top1Absent and Top5Absent.

6.2 Experimental Design

6.2.1 Independent and Dependent Variables

Independent variables: search engines (Google, Bing, CBing and Baidu)

Dependent variable: Rate Top1Absent hourly ROCOA, Rate Top5Absent hourly ROCOA.

In this experiment, all the search engines used the same 500 original queries.

6.2.2 Experimental Procedures

At first, 500 English words were randomly selected from an English dictionary mentioned in Section 3.1.1 and they are attached in Appendix B. These 500 words were regarded as the original queries, while the follow-up queries were the original query to which was added the domain names of the first five results of the original query. For example, an original query is a word "*A*" whose first five results' top-level domain names are ".com" , ".edu", ".gov", ".au" and ".net", so the five follow-up queries are *"A" site:.com*, *"A" site:.edu*, *"A" site:.gov*, *"A" site:.au* and *"A" site:.net.* If the top 50 results of the first follow-up query *"A" site:.com* do not include the first result of the original query, then a Top1Absent anomaly has occurred.

If any of the following occurs, then a Top5Absent anomaly has occurred:

1. Top1Absent (that is, the top 50 results of the first follow-up query *"A" site:.com* do not include the first result of the original query)

2. The top 50 results of the second follow-up query *"A" site:.edu* do not include the second result of the original query.

3. The top 50 results of the third follow-up query *"A" site:.gov* do not include the third result of the original query.

4. The top 50 results of the fourth follow-up query *"A" site:.au* do not include the fourth result of the original query.

5. The top 50 results of the fifth follow-up query *"A" site:.net* do not include the fifth result of the original query.

Figure 6.1 is a example of Bing Tob1Absent. As stated in Section 3.1, all advertisements were removed from the search results, the first result of the original query was "Chili's" (www.chilis.com). The top 50 results of the follow-up query did not include this result, so there was a Top1Absent anomaly. Because it is infeasible to include all the top 50 results of the follow-up query in the figure, we only show the first two results in order to help the author to explain the idea. Because Top1Absent anomaly occurred, Top5Absent anomaly also occurred.

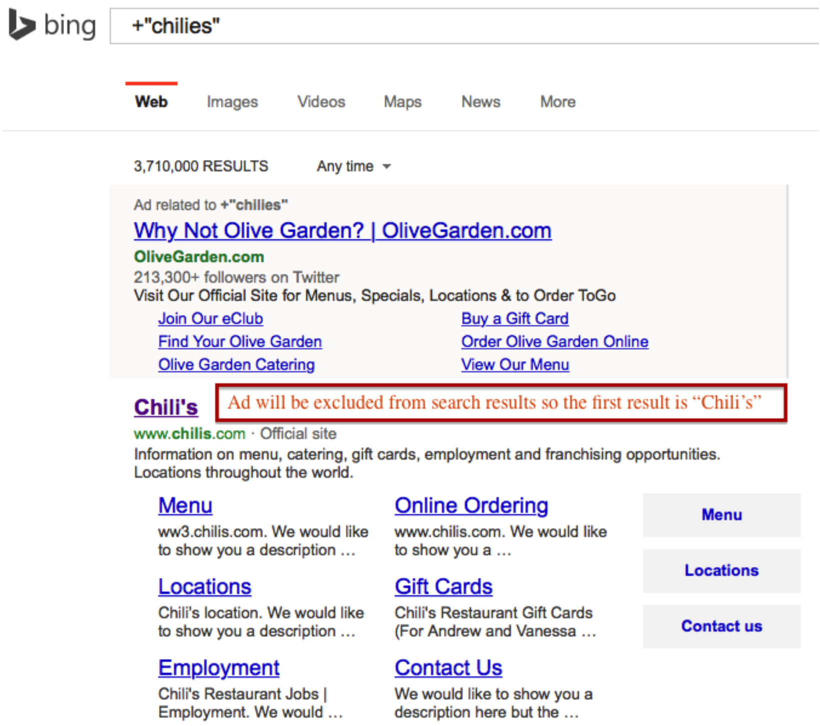
For Bing, CBing and Baidu, all these 500 words were tested every hour, while for Google these 500 words were tested every three hours because the resource was limited. Because every original query had at most five corresponding follow-up queries, at most 2500 test case pairs were tested in every batch. For Bing, CBing and Baidu a batch was tested in one hour while for Google it was tested every three hours. Table 6.1 shows the number of test case pairs tested in this experiment.

Table 6.1: The number of test case pairs in the experiment Ranking Dropping with domain

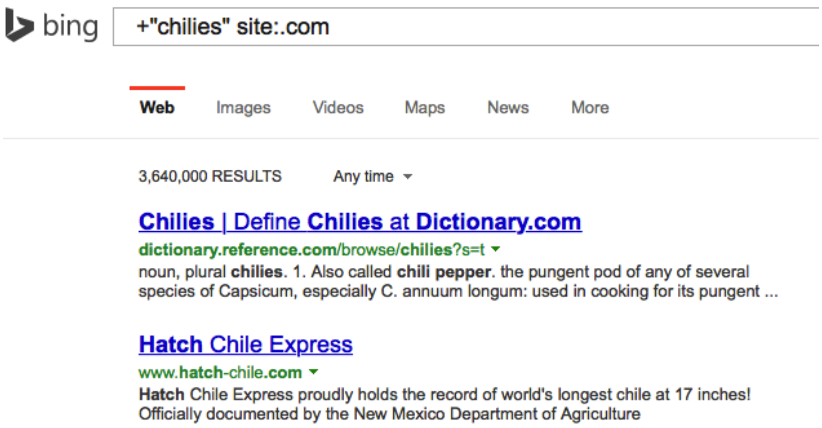
Search Engine	Test case pairs per hour (approximate)	Hours	Total test case pairs (approximate)
Google	2500	353	882,500
Bing	2500	353	882,500
CBing	2500	353	882,500
Baidu	2500	353	882,500

6.3 Threats to Validity

The main concern with validity in the experiment was the correctness of the MRs Top1Absent and Top5Absent. All the advertisements in advertisement sections of search engines were not included in search results. Also, the time period between the original query and the follow-up query is very short, so when a result dropped, then we consider it is an anomaly rather than database updating. Of course, search engines might put the advertisements in the main section of search results same as normal results without telling users, but they should also follow the rules in their user manuals; otherwise, it was reasonable for users to argue there were anomalies in the search engines. If a result ranked as the i^{th} ($i \in N$, $0 < i < 6$) result of the query "A",

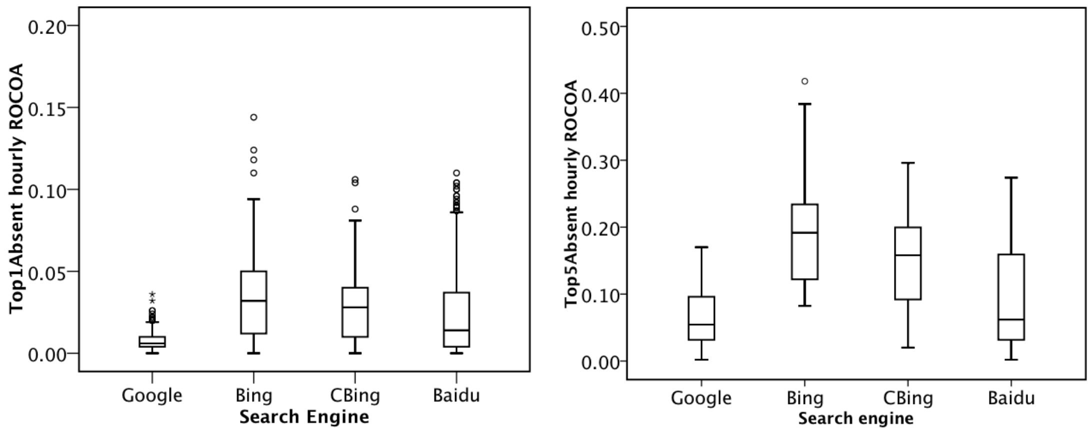


(a)Original query: the first result is "Chili's" (www.chilis.com) because advertisement will be removed from search results



(b)Follow-up query: First result of original query in out of top 50, but it is infeasible to include all the top 50 results in this figure

Figure 6.1: An example of Bing Top1Absent on 19 January 2015 (a) Original query; (b) Follow-up query.



(a)Top1Absent (b)Top5Absent

Figure 6.2: No ranking drop with Domain hourly ROCOA of the four search engines: (a) Top1Absent hourly ROCOA of the four search engines; (b) Top5Absent hourly ROCOA of the four search engines.

that means the search engine considered it to have a higher ranking than any other results that were ranked after it (including of course the results from the same domain with the first result of query "A"). After adding the domain name to the query, only the results from that domain will be returned, therefore the i^{th} result should be ranked as $(i - n)^{th}$ result where $n \in N$, $0 \leq n < i$. Obviously, if the result was ranked out of top 50, there should be an anomaly.

6.4 Experimental Results

The results of Top1Absent and Top5Absent are shown in Figure 6.2 which only shows the results of those hours when all four search engines were tested. A one-way ANOVA was conducted to compare the differences between the search engines on Top1Absent and Top5Absent. With the Top1Absent and Top5Absent, there were significant differences between Google, Baidu, Bing, and CBing at the 0.01 level with $F(3, 1408) = 106.134$, $F(3, 1408) = 263.599$, respectively. As Table 6.4 shows, post-hoc comparisons using the Games-Howell method indicated that the Top1Absent of

Table 6.2: Multiple comparisons of Top1Absent and Top5Absent hourly ROCOA, using the Games-Howell procedure. The mean differences in highlighted cells are significant at 0.05 level

Multiple Comparisons			
Games-Howell			
(I) Search Engine	(J) Search Engine	Mean Difference (I-J)	Sig.
Top1Absent hourly ROCOA			
Google	Bing	-0.0271	<0.001
Google	CBing	-0.0198	<0.001
Google	Baidu	-0.0199	<0.001
Bing	CBing	0.0073	<0.001
Bing	Baidu	0.0072	0.002
CBing	Baidu	<0.0001	1
Top5Absent hourly ROCOA			
Google	Bing	-0.1207	<0.001
Google	CBing	-0.0879	<0.001
Google	Baidu	-0.0330	<0.001
Bing	CBing	0.0327	<0.001
Bing	Baidu	0.0877	<0.001
CBing	Baidu	0.0549	<0.001

Google was the smallest (M = 0.0071, SD = 0.0052) of the four search engines, and it was significantly smaller than the other search engines with $p < 0.001$. However, Bing received the largest Top1Absent value (M=0.0342, SD = 0.0242), which was significantly larger than that of the other search engines with $p < 0.001$, although there was no difference between Top1Absent of CBing (M = 0.0269, SD = 0.0176) and Baidu (M=0.0269, SD = 0.0297). The Top5Absent of any two of the four search engines differed significantly; indeed of the four search engines, Google had the smallest Top5Absent (M=0.0649, SD = 0.0395) and Bing had the largest Top5Absent rate (M=0.1856, SD = 0.0677) whereas the Top5Absent rate of Baidu (M = 0.0977, SD = 0.0766) was significantly smaller than CBing (M=0.1529, SD=0.0607) with $p < 0.001$.

Chapter 7

Additional Findings

7.1 Are Search Results Biased by Search Engine for Commercial Interest?

Research question: Will search engine manipulate search results for commercial interest?

Because different users may use keywords with different commercial value, I will investigate whether search results biased by search engine for commercial interest [57]. In this section, correlations between metrics of MRs and the *average number of advertisements per query* was analysed. The *Average number of Advertisements Per Query (AAPQ)* can be given by Equation 7.1.

$$AAPQ = \frac{Total\ the\ number\ of\ ads\ in\ one\ hour}{Total\ number\ of\ queries\ of\ that\ hour} \tag{7.1}$$

Spearman's rank correlation is used in this thesis because it is a non-parametric method and, hence, it is universally applicable. Also, we are trying to find monotonic relationship between our variables and Spearman's rank correlation is robust to outliers.

The correlation between MPSite hourly ROCOF and AAPQ of Bing English was

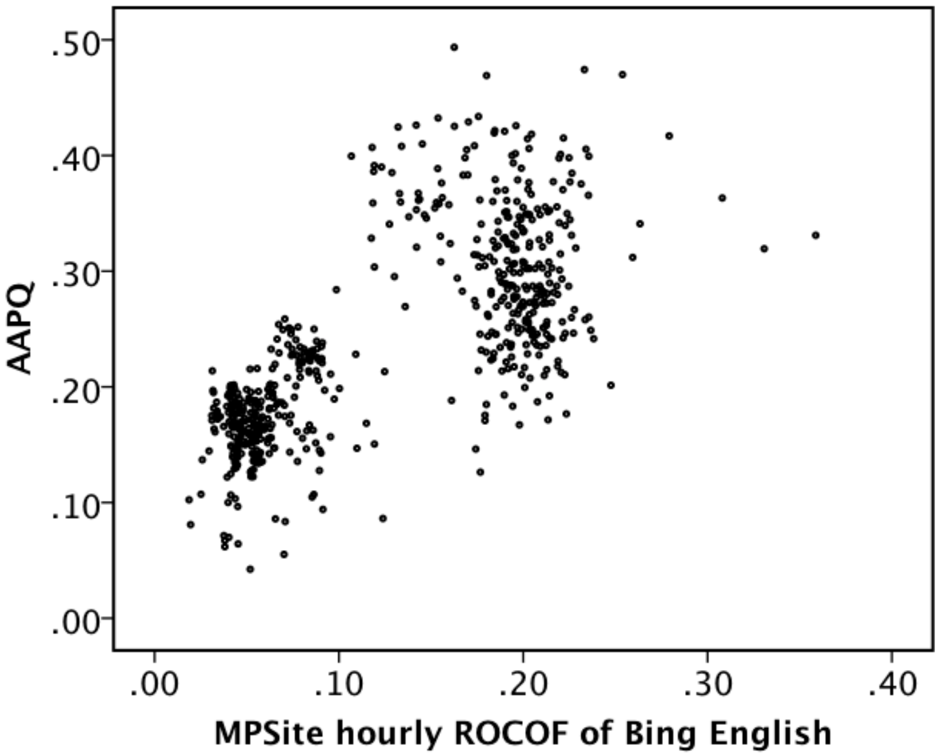
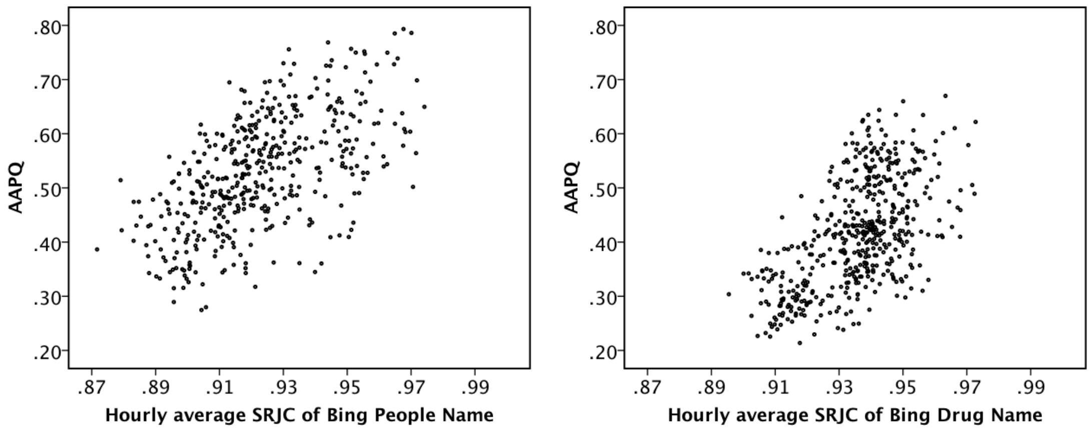


Figure 7.1: Correlation of MPSite hourly ROCOF and AAPQ of Bing English (Spearman's rho: r(604) = .724, p <0.001)

analysed and the result is shown in Figure 7.1. The result shows that the two variables of Bing are strongly correlated, r(604) = .724, p <0.001 at the 2-tailed 0.01 level. That is to say, the increases in the reliability of Bing on the MPSite were correlated with the decrease in the average number of advertisements per query, although the correlations between the two variables of the other three search engines were weak.

The correlation between the hourly average SRJC and the AAPQ was analysed and the results are shown in Figure 7.2. The result shows that in Bing the hourly average SRJC of person names and drug names were moderately correlated with the AAPQ with r(452) = .603, p <0.001 and r(496) = .573, p <0.001 at 2-tailed 0.01 level, respectively. That is to say, the increases in the page retrieval capability of Bing on metric hourly average SRJC on person names and drug names were correlated with the increases in the AAPQ value. However, no correlation was found between the hourly



(a)Bing hourly average SRJC of people names: (Spearman's rho: r(452) = .603, p <0.001)

(b)Bing hourly average SRJC of drug names: (Spearman's rho: r(496) = .573, p <0.001)

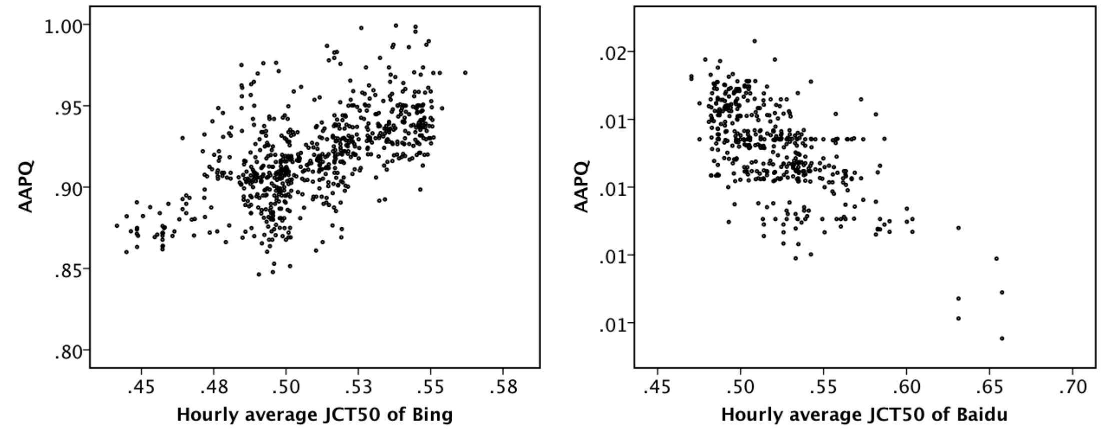
Figure 7.2: Correlation between Bing hourly average SRJC and AAPQ:

average SRJC of the other three search engines and their AAPQ value.

The correlation between the hourly average JCT50 and the AAPQ of Baidu are shown in Figure 7.3. Moderate correlations between the two valuables were found in Bing and Baidu, i.e., r(727) = .630, p <0.001 and r(446) = -.586, p <0.001 respectively. The result indicated that the increases in the ranking consistency of Bing on the hourly average JCT50 was correlated with the increases in AAPQ value. However, there is a negative correlation between the ranking consistency of Baidu on the metric hourly average JCT50 and the AAPQ value. A correlation does not necessarily mean a causal relation and an investigation into the causes for these correlations is beyond the scope of this thesis.

The correlations between the MRs of No Ranking Drop with Domain and the AAPQ are shown in Figure 7.4. In Baidu, there was a strong negative correlation between the AAPQ and Top1Absent with r(189) = -.794, p <0.001. Similarly, there was a strong negative correlation between the AAPQ and Top5Absent with r(189) = -.750, p <0.001. Overall, the performance of Baidu on Top1Absent and Top5Absent were positively correlated with the AAPQ.

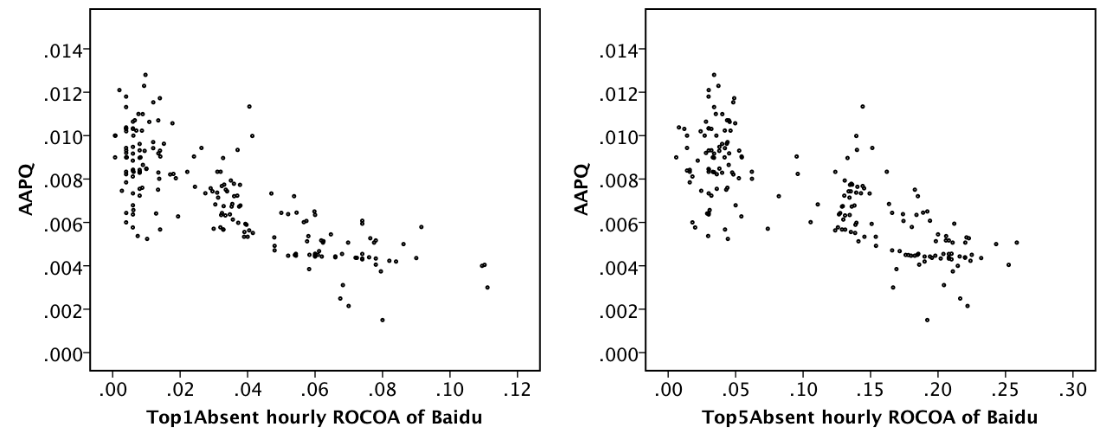
From the correlations listed above, we can see that the majority of metrics did



(a)Correlation between hourly average JCT50 and AAPQ of Bing: r(727) = .630, p <0.001

(b)Correlation between hourly average JCT50 and AAPQ of Baidu: r(446) = -.586, p <0.001

Figure 7.3: Correlation between hourly average JCT50 and AAPQ.



(a)Correlation between Top1Absent hourly ROCOA of Baidu and AAPQ: (Spearman's rho: r(189) = -.794, p < 0.001)

(b)Correlation between Top5Absent hourly ROCOA of Baidu and AAPQ: (Spearman's rho: r(189) = -.750, p < 0.001)

Figure 7.4: Correlation between Top1Absent and Top5Absent hourly ROCOA of Baidu and AAPQ.

not have correlations with the number of advertisements. The quality of a search engine may has positive correlation with the number of advertisements on one metric but has negative correlation on other metrics. For instance, in Bing, the performance on MPSite was negatively correlated with the number of advertisements while the performance on SwapJD was positively correlated with the number of advertisements. Similarly, in Baidu, the performance on SwapJD was negatively correlated with the number of advertisements while the performance on Top1Absent and Top5Absent were positively correlated with the AAPQ, because a higher Top1Absent or Top5Absent hourly ROCOA value means a worse performance. Correlation does not mean causal relation and the reason why these correlations exist is unknown to us. In conclusion, we do not find any obvious evidence of search engine manipulating search results for commercial interests.

7.2 Correlations between MRs

The seven metamorphic relations were from different aspects and used different methods to validate the performance of the four search engines. Is there any relationship between the metrics of MRs so that we can predict the scores of some metrics of MRs using the scores of some others? If we can, we do not need to use all the metrics every time when we evaluate search engines, especially when time is limited. This section seeks to discover the correlations between different metrics of MRs of each search engine, and in order to do so, we only selected those data from those hours when both of the two metamorphic relations were tested. In all the four search engines, Top1Absent and Top5Absent have strong correlations, but we do not report them because the correlations are expected. We did not find any correlation between different metamorphic relations in Google and Baidu while some correlations were found in Bing and CBing.

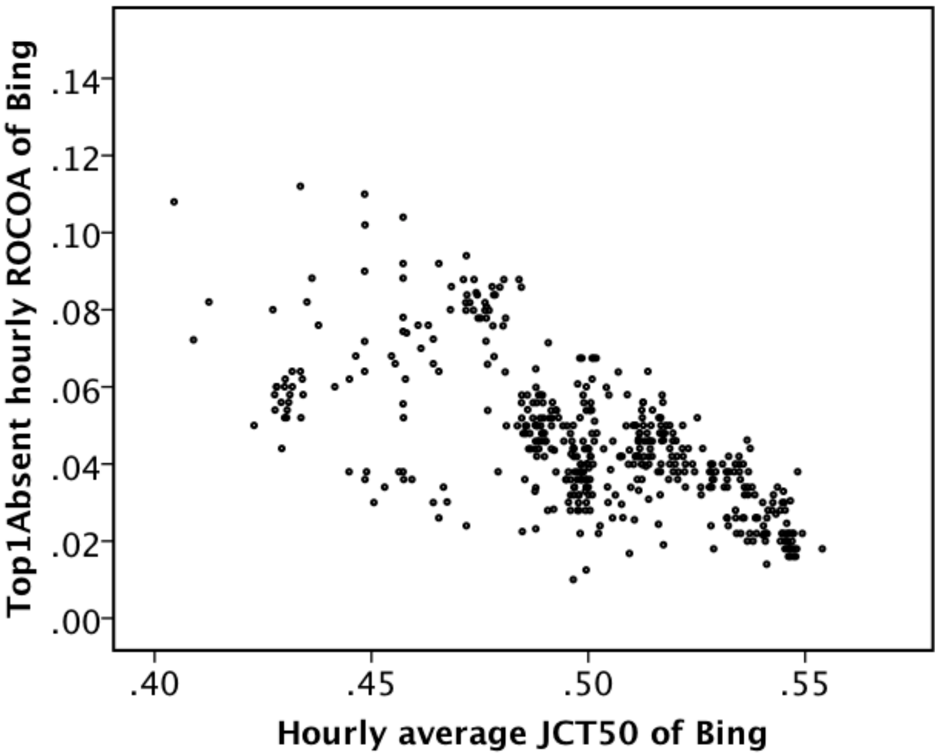
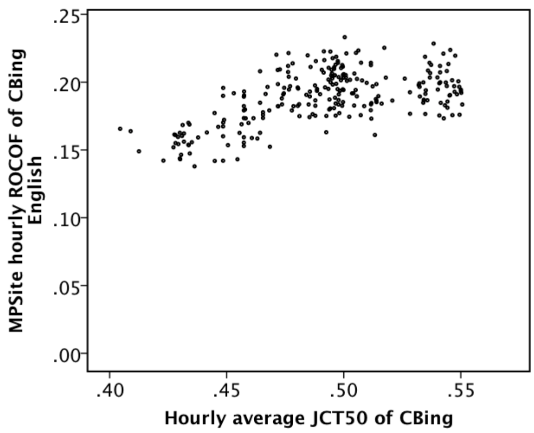


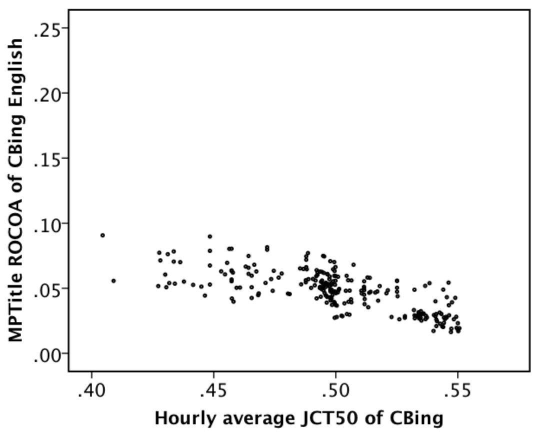
Figure 7.5: Correlation between hourly average JCT50 and Top1Absent hourly ROCOA of Bing: (Spearman's rho: $r(456) = -.643$, $p < 0.001$)

Figure 7.5 shows the correlation between hourly average JCT50 and Top1Absent hourly ROCOA of Bing. It can be seen from the figure that there are moderate correlation between hourly average JCT50 of Universal SwapJD and the Top1Absent hourly ROCOA was found in Bing with $r(456) = -.643$, $p < 0.001$. A smaller Top1Absent hourly ROCOA value indicates a better performance while a smaller hourly average JCT50 value indicates a worse performance. Therefore, this result shows that the quality of Bing in Top1Absent and SwapJD have a positive correlation.

In Figure 7.6, a moderate correlation was found between hourly average JCT50 of Universal SwapJD and MPSite hourly ROCOF in CBing with $r(256) = .507$, $p < 0.001$ and a strong correlation between the hourly average JCT50 of Universal SwapJD and MPTitle hourly ROCOA were found in CBing with $r(241) = -.727$, $p < 0.001$. It can be seen that the MPSite hourly ROCOF of CBing and the score of CBing on the



(a)Correlation between hourly average JCT50 and MPSite hourly ROCOF of CBing: (Spearman's rho: $r(256) = .507$, $p < 0.001$)



(b)Correlation between hourly average JCT50 and MPTitle hourly ROCOA of CBing: (Spearman's rho: $r(241) = -.727$, $p < 0.001$)

Figure 7.6: Correlations between MRs of CBing.

metric hourly average JCT50 of Universal SwapJD had a positive correlation while the score of CBing on metric hourly average JCT50 and the MPTitle hourly ROCOA had a negative correlation. For MPSite hourly ROCOF and MPTitle hourly ROCOA, a smaller value indicates a better performance. However, a smaller hourly average JCT50 value indicates a worse performance. Therefore, we can get a conclusion that the quality of CBing in SwapJD was negatively correlated with its quality in MPSite but positively correlated with its quality in MPTitle.

In conclusion, some correlations are found in only three cases,, but the reason for these correlations are unknown. This means that, in most situations, there is no correlation among the MRs. Therefore, the search engine's performance against one MR cannot imply its performance against another MR. In other words, all MRs identified in this thesis are necessary and there is no redundancy.

Chapter 8

Conclusion and Future Work

Search engine evaluation is hard because there are no test oracles. This thesis used the concept of MT in evaluating search engine. We proposed seven novel MRs for the evaluation of search engine. Also empirical study was conducted to show our MRs are efficient in search engine evaluation.

This thesis has made the following contributions:

1. We applied the metamorphic testing method to search engines. This enables the detection of failures or anomalies despite the oracle problem. As a result, our method also enables conventional reliability metrics to be applied to search engines.

2. We created seven novel MRs for the evaluation of search engine.

3. Using the seven MRs, we conducted a large-scale empirical evaluation on the web page retrieval and ranking qualities of four major search engines. The quality scores of these four search engines were compared; many results are statistically significant.

4. We analysed the correlations between the search quality and some other factors such as advertisements, query languages, nature of query keywords, and search domains. These results are useful for both developers and users to understand the search engine behaviour and provide hints for developers to locate potential faults and to better tune the search engines.

5. We also analysed the correlations between the fault-detection effectiveness of different metamorphic relations. For instance, a strong correlation was found between Universal SwapJD and MPSite in CBing. This result suggests that it may not be necessary to test all metamorphic relations so that testing cost can be saved without affecting the fault-detection effectiveness. More research on this topic will be conducted in future study.

Future research will include the identification and optimization of a larger set of MRs in order to find more problems in the search engines. Apart from this, future research will also include a study on the effect of human activities (for example, weekends and public holidays) on the performance of search engines. We will also include other types of search in our study, such as images search, video search, and map search.

Appendix A

Names used in MPReverseJD experiment

This section includes all the names used in MPReverseJD experiment, namely 200 person names, 200 company names and 200 drug names.

200 person names	200 company names	200 drug names
Marilyn Monroe	Royal Dutch Shell	Abacavir
Mother Teresa	Exxon Mobil	Acebutolol
John F. Kennedy	Wal-Mart Stores	Acetazolamide
Martin Luther King	BP	Acyclovir
Nelson Mandela	Sinopec Group	Albendazole
Winston Churchill	China National Petroleum	Amantadine
Bill Gates	State Grid	Amikacin
Muhammad Ali	Chevron	Amiloride
Mahatma Gandhi	ConocoPhillips	Aminoidouridine
Margaret Thatcher	Toyota Motor	Amlodipine
Charles de Gaulle	Total	Amphotericin
Christopher Colombus	Volkswagen	Ampicillin
George Orwell	Japan Post Holdings	Amprenavir
Charles Darwin	Glencore International	Anagrelide
Elvis Presley	Gazprom	Arteflene
Albert Einstein	E.ON	Artemether
Paul McCartney	ENI	Artemisinin
Plato	ING Group	Aspirin
Queen Elizabeth II	General Motors	Atenolol
Queen Victoria	Samsung Electronics	Atorvastatin
John M Keynes	Daimler	Atovaquone
Mikhail Gorbachev	General Electric	Azithromycin
Jawaharlal Nehru	Petrobras	Aztreonam
Leonardo da Vinci	Berkshire Hathaway	Bacitracin
Louis Pasteur	AXA	Becampicillin
Leo Tolstoy	Fannie Mae	Benzepril
Pablo Picasso	Ford Motor	Betaxolol
Vincent Van Gogh	Allianz	Bezafibrate
Franklin D. Roosevelt	Nippon Telegraph & Telephone	Bisoprolol
Pope John Paul II	BNP Paribas	Bretylium
Neil Armstrong	Hewlett-Packard	Bromodeoxyuridine
Thomas Edison	AT&T	Bumetanide
Rosa Parks	GDF Suez	Butenafine
Aung San Suu Kyi	Pemex	Candesartan
Lyndon Johnson	Valero Energy	Captopril
Ludwig Beethoven	PDVSA	Carvedilol
Oprah Winfrey	McKesson	Caspofungin
Indira Gandhi	Hitachi	Cefaclor
Eva Peron	Carrefour	Cefadroxil
Benazir Bhutto	Statoil	Cefamandole
Desmond Tutu	JX Holdings	Cefixime
Dalai Lama	Nissan Motor	Cefoperazone
Walt Disney	Hon Hai Precision Industry	Cefoxitin
Peter Sellers	Banco Santander	Cefprozil
Barack Obama	EXOR Group	Ceftazidime
Malcolm X	Bank of America	Ceftibuten
J.K.Rowling	Siemens	Ceftriaxone
Richard Branson	Assicurazioni Generali	Cephalothin
Pele	Lukoil	Cholestipol
Jesse Owens	Verizon Communications	Cilistatin
Ernest Hemingway	J.P. Morgan Chase & Co.	Cinoxacin
John Lennon	Enel	Ciprofibrate

Henry Ford	HSBC Holdings	Ciprofloxacin
Haile Selassie	Industrial & Commercial Bank of China	Clindamycin
Joseph Stalin	Apple	Clofazimine
Lord Baden Powell	CVS Caremark	Clonidine
Michael Jordon	International Business Machines	Clopidogrel
George Bush jnr	Crédit Agricole	Clotrimazole
V.Lenin	Tesco	Cloxacillin
Osama Bin Laden	Citigroup	Cycloserine
Fidel Castro	Cardinal Health	Dalfopristin
Oscar Wilde	BASF	Dapsone
Coco Chanel	UnitedHealth Group	Daptomycin
Amelia Earhart	Honda Motor	Dichlorphenamide
Adolf Hitler	SK Holdings	Digitoxin
Mary Magdalene	Panasonic	Diltiazem
Alfred Hitchcock	Société Générale	Dirithromycin
Michael Jackson	Petronas	Dobutamine
Madonna	BMW	Doxazosin
Mata Hari	ArcelorMittal	Enalapril
Cleopatra	Nestlé	Enoxacin
Emmeline Pankhurst	Metro	Ertapenem
Ronald Reagan	Électricité de France	Erythromycin
Lionel Messi	Nippon Life Insurance	Esmolol
Babe Ruth	Kroger	Ethambutol
Bob Geldof	Munich Re Group	Ethoxazolamide
Leon Trotsky	China Construction Bank	Felodipine
Roger Federer	Costco Wholesale	Fenofibrate
Sigmund Freud	Freddie Mac	Flecainide
Woodrow Wilson	Wells Fargo	Fluconazole
Mao Zedong	China Mobile Communications	Fosinopril
Katherine Hepburn	Telefónica	Furazolidone
Audrey Hepburn	Indian Oil	Gatifloxacin
David Beckham	Agricultural Bank of China	Gemfibrozil
Tiger Woods	Peugeot	Gentamicin
Usain Bolt	Procter & Gamble	Grepafloxacin
Bill Cosby	Sony	Griseofulvin
Carl Lewis	Banco do Brasil	Guanethidine
Prince Charles	Deutsche Telekom	Hydrochlorothiazide
Jacqueline Kennedy Onassis	Repsol YPF	Hydralazine
C.S. Lewis	Noble Group	Ibutilide
Billie Holiday	Archer Daniels Midland	Imipenem
J.R.R. Tolkien	Bank of China	Indapamide
Virginia Woolf	AmerisourceBergen	Irbesartan
Billie Jean King	PTT	Isoniazid
Kylie Minogue	Meiji Yasuda Life Insurance	Isoproterenol
Anne Frank	Toshiba	Isradipine
Emile Zatopek	Deutsche Post	Itraconazole
Lech Walesa	Reliance Industries	Kanamycin
Christiano Ronaldo	China State Construction Engineering	Ketoconazole
Gunnar Myrdal	China National Offshore Oil	Labetalol
William Faulkner	INTL FCStone Inc.	Levofloxacin
John Dos Passos	Groupe BPCE	Lidocaine
George VI	Deutsche Bank Aktiengesellschaft	Linezolid
Aldous Huxley	Vodafone Group Plc	Lisinopril

Reinhold Niebuhr	Marathon Petroleum	Loracarbef
Hu Shih	Walgreen Co.	Lovastatin
Ho Chi Minh	BHP Billiton Limited	Methazolamide
John Foster Dulles	American International Group,Inc.	Mezlocillin
Rupert Brooke	Robert Bosch GmbH	Minoxidil
Van Wyck Brooks	China Railway Construction	Moxifloxacin
Ezra Pound	China Railway Group	Mupirocin
Harry Truman	Sinochem Group	Nafcillin
William Carlos Williams	MetLife	Neomycin
Jacques Derrida	Mitsubishi	Nicardipine
Douglas MacArthur	The Home Depot	Norfloxacin
Albert Einstein	Hyundai Motor Company	Nystatin
Carl Sandburg	Medco Health Solutions	Ofloxacin
Isadora Duncan	Microsoft	Oxacillin
Piux XII	Target	Oxytetracycline
Thomas Mann	Barclays Plc	Penicillin
Winston Churchill	ThyssenKrupp AG	Paromomycin
Al Smith	The Boeing Company	Penbutolol
Sri Aurobindo	RWE Aktiengesellschaft	Polythiazide
Cordell Hull	Pfizer Inc.	Prazosin
Frank Norris	The Tokyo Electric Power Company	Pronotosil
Andre Gide	China Life Insurance (Group) Company	Quinapril
William Allen White	SAIC Motor Limited	Quinethazone
Arnold Bennett	Lloyds Banking Group plc	Quinidine
Ramsay MacDonald	Mitsui	Quinupristin
Theodore Roosevelt	PepsiCo	Rifampin
John Dewey	AEON	Rifapentine
Jane Addams	United States Postal Service	Reserpine
Rabindranath Tagore	Banco Bradesco S.A.	Ramipril
Edward Grey	Rosneft Oil Company	Rosuvastatin
David Lloyd George	Johnson & Johnson	Simvastatin
Max Weber	Unilever N.V./ Unilever PLC	Sorbitol
Rudyard Kipling	State Farm Insurance Cos.	Sparfloxacin
George Bancroft	Dongfeng Motor Group	Spectinomycin
Brigham Young	The Royal Bank of Scotland Group plc	Sulfacetamide
Victor Hugo	Mitsubishi UFJ Financial Group	Tacrolimus
Ralph Waldo Emerson	The Dai-ichi Life Insurance Company	Tamoxifen
George Sand	POSCO	Tapentadol
William Lloyd Garrison	Dell Inc.	Tazarotene
John Stuart Mill	Aviva plc	Tazobactam
Louis Agassiz	Groupe Auchan	Tegaserod
Napoleon III	WellPoint	Telavancin
Abraham Lincoln	Seven & I Holdings	Telbivudine
Leo XIII	China Southern Power Grid	Telithromycin
Horace Greeley	Rio Tinto Group	Telmisartan
Charles Dickens	Caterpillar Inc.	Temazepam
Henry Ward Beecher	The Dow Chemical Company	Temozolomide
Charles Reade	Novartis AG	Temsirolimus
Anthony Trollope	Renault S.A.	Tenecteplase
Russell Sage	Vale S.A.	Teniposide
Henry David Thoreau	Bunge Limited	Tenofovir
Karl Marx	Compagnie de Saint-Gobain	Terazosin
George Eliot	Prudential plc	Terbinafine

Herbert Spencer	United Technologies	Terbutaline
Mary Baker Eddy	UniCredit Group	Terconazole
Matthew Arnold	China FAW Group	Terfenadine
Goldwin Smith	Fujitsu Limited	Terpin Hydrate
Stonewall Jackson	Comcast	Testosterone
Bayard Taylor	Marubeni	Urea
Walter Bagehot	China Minmetals	Urokinase
Charles Eliot Norton	Kraft Foods Inc.	Ursodiol
George Meredith	Wesfarmers Limited	Ustekinumab
Carl Schurz	Itochu	Valacyclovir
Emily Dickinson	Intel	Valdecoxib
Sitting Bull	Nokia	Valerian
Leslie Stephen	Woolworths Limited	Valganciclovir
Edwin Booth	United Parcel Service	Valproic Acid
William Morris	Zurich Insurance Group Ltd.	Valrubicin
Mark Twain	Deutsche Bahn AG	Valsartan
Bret Harte	Nippon Steel	Vancomycin
Grover Cleveland	Manulife Financial	Varenicline
John Morley	CNP Assurances S.A.	Vasopressin
Henry George	Vinci	Vecuronium
Crazy Horse	Best Buy Co.	Venlafaxine
Edward VII	LyondellBasell Industries N.V.	Verapamil
Alfred Marshall	Banco Bilbao Vizcaya Argentaria, S.A.	Verteporfin
Henry James	Bayer AG	Vidarabine
Anatole France	Saudi Basic Industries	Vigabatrin
Elihu Root	SSE PLC	Vinblastine
Buffalo Bill	Lowe's Companies	Vincristine
Ellen Terry	Sumitomo Mitsui Financial Group	Vinorelbine
Grant Allen	Roche Holding Ltd.	Warfarin
Edmund Gosse	Intesa Sanpaolo S.p.A.	Zafirlukast
Robert Louis Stevenson	CITIC Group	Zalcitabine
Oliver Lodge	Prudential Financial	Zaleplon
Brander Matthews	LG Electronics Inc.	Zanamivir
Cecil Rhodes	Baosteel Group	Ziconotide
Josiah Royce	TNK-BP International Ltd.	Zidovudine
Pius XI	Idemitsu Kosan	Zileuton
Nawaz Sharif	Sanofi	Ziprasidone
Clarence Thomas	Veolia Environnement SA	Zoledronic Acid
Bill Clinton	Hyundai Heavy Industries	Zolmitriptan
Daniel Ortega	Credit Suisse Group AG	Zolpidem
Terry Eagleton	China North Industries Group Corporation	Zonisamide
Bob Dylan	Amazon.com Inc.	Zuclopenthixol

Appendix B

Original queries of No Ranking Drop with Domain

The 500 original queries were randomly selected from the English dictionary [51]. In these 500 queries, some words may have spelling mistakes, which is appropriate in the experiment because real users often make some spelling mistakes when they are typing queries to search engines.

The 500 original queries used in No Ranking Drop experiment:

kitten	skimo	intermitter	barque
eat	cohort	deerskin	owning
wallaby	commandeer	stapedes	mimesises
handfast	krill	discounters	parasols
depravedness	flirtations	requin	nival
contemptibility	wakers	region	embosked
preeminent	isms	relegation	hoke
deficiency	hopelessness	recruit	dabbling
pluralist	windburnt	electrocardiograms	ruffed
establisher	enlisting	incisively	outspoke
quadrate	hobby	shadoof	stinky
transposition	elute	carcanets	erns
nonsuiting	tyrannize	barmaids	operceles
stormiest	negus	bedamns	hajjis
endurances	dictating	saucers	landslide
menhirs	crouched	hucksterisms	swisses
executers	minders	incommodes	sippers
prevues	pinocles	pogroms	ciphony
uplighting	modernizes	jalaps	nided
irritably	identifiably	pistons	churr
backslides	incuse	sassier	classes
actin	mezcal	studded	seizure
biogens	chilies	firerooms	almners
butchering	slipsole	buirdly	sternest
wampuses	pishing	foreshown	federation
timberline	beneficed	flotsams	trigness
coalsheds	signori	orphanhood	metabolisms
mezcals	abstricting	scuds	velure
isopleths	undereate	aircrew	detraction
trenail	subtones	countering	garishly
jongleur	restaurateur	hubbies	colluders
scarabs	doty	swanherds	levelness
scraichs	ghoulishness	dejected	kaliphs
prosthetically	engrain	feminise	incitation
gossan	cork	osier	snitched
fishless	seasickness	horsehide	renvoi
credits	cundum	machrees	gravy
proscriptive	reverted	bailsmen	farm
conniver	juga	rasps	moneybags
brays	chrismal	decliner	interjections
bighting	waggoner	retracting	sequelae
bailiwicks	cesarian	reexpelling	cacique
procreations	parritches	concatenation	pedagogics
sootiness	answerer	colleted	eelworms
filmdom	mikado	panned	luckily

gaging
inconsiderable
grapevines
newsmen
materializations
formers
chaldron
slick
smolt
platters
solonets
buttressed
fishbowls
stilbites
zabaione
juggernauts
christened
perjures
brio
ustulate
spooney
sanitizations
castrations
west
ineradicably
ethylating
plover
quicks
pinup
fingernail
rental
triode
benzoles
dashy
agenes
inextinguishable
deliveries
infusible
haver
deediest
excel
capitulate
bechalks
trochaic
fomentation
grumes

translucence
unplug
carroms
gradualists
rejudge
yank
upbow
placentations
colas
anurias
demagogs
markhors
cacti
demurely
popularizes
sic
mudslingings
crusado
valiancies
dyspnoic
cotqueans
titrators
antiperspirant
wames
murrain
dulcineas
soberized
savvying
coconspirators
sinister
blether
boche
hurricanes
dill
depone
areae
triplex
initiation
monomial
insensitiveness
tynes
yoghourt
pacing
orators
filcher
requiting

coplotted
materiel
frizzer
retrospect
rhumba
overvote
recharging
solubilities
psychologist
sewer
paludism
mameyes
weighted
inducer
curvy
hillocks
simoom
embezzles
soliloquizer
spastically
temporized
miscall
betokens
gunboat
manacling
gems
dreks
briars
exorbitant
redargues
misprizes
downturn
sleevelet
bale
egad
confederated
porterhouses
contrarieties
forested
campo
dislodge
smuggling
isolated
idiolects
yeaned
uncharacteristic

semiweekly
manicuring
modelling
scopulae
nappy
orthodontist
habit
weekday
namelessly
venules
criterion
quarantining
relate
osseins
horsewhips
sloughy
him
refrangible
peddles
senility
sains
anatomized
shorl
poortiths
bleaches
nitroso
nonrecurring
moseying
slanging
moving
camporees
interfered
ragouted
patienter
overstocking
cochair
decelerator
printings
muckluck
lampooned
quiz
foreshadower
oxidation
milliards
pimpled
blooded

emancipation	zooks	childless	legislator
designing	dimplier	pachysandra	misused
pluvial	veratrin	reignite	decentralization
consignable	upbearing	placing	gum
fondus	seely	hepatitides	uncini
maestoso	kousso	luminous	slenderest
seediest	mastitic	minimizer	salability
logogriph	sonnetized	tinplate	carats
cooing	quey	waged	bedighting
massa	preflight	microbuses	objectionably
sprattled	sortable	threading	oxcarts
sassaby	finialed	inchmeal	epicure
mangling	pasquils	succeeding	meaty
boarfishes	ripieno	shutoffs	sourdines
pointless	congeal	posterns	suburban
panegyrics	evocator	flited	intenser
forecloses	symbolist	stricter	imperilments
batons	tenderizes	disharmonies	eel
pollen	tipplers	arb	teleran
preshowing	changeling	millihenrys	politicized
arrives	marblings	bads	malamutes
intimas	zippy	unpacks	hamulose
prattler	synchrotron	objurgate	reship
hyperventilation	hammertoe	supines	carburetors
listlessness	deformer	thionic	scarier
insisted	gabelles	lands	mesonic
interjecting	sarsaparillas	whitey	crowns
downtown	vapidness	interlace	mouchoir
depend	thefts	stencilers	seduces
reverberation	exasperation	tyramines	envelopes
dripping	brutalizes	relics	hypothec
munched	papillary	prochein	kneepads
larval	typicality	thunk	upheld
installed	perverse	delusory	anabatic

References

[1] C. Ghezzi, M. Jazayeri, and D. Mandrioli. *Fundamentals of Software Engineering*. Prentice Hall, Upper Saddle River, New Jersey, 2003.

[2] E. J. Weyuker. On testing non-testable programs. *The Computer Journal*, 25(4): 465–470, November 1982.

[3] L. I. Manolache and D. G. Kourie. Software testing using model programs. *Software: Practice and Experience*, 31(13):1211–1236, November 2001. ISSN 1097-024X. doi: 10.1002/spe.409.

[4] I. Soboroff. Dynamic test collections: Measuring search effectiveness on the live Web. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'06)*, pages 276–283, New York, NY, 2006. ACM Press. ISBN 1-59593-369-7. doi: 10.1145/1148170.1148220.

[5] R. Mall. *Fundamentals of Software Engineering*. PHI Learning Private Limited, 3 edition, 2004.

[6] C. W. Cleverdon, J. Mills, and M. Keen. Factors determining the performance of indexing systems. Technical report, Cranfield University, 1966.

[7] Z. Q. Zhou, S. Zhang, M. Hagenbuchner, T. H. Tse, F.-C. Kuo, and T. Y.

Chen. Automated functional testing of online search services. *Software Testing, Verification and Reliability*, 22(4):221–243, June 2012.

[8] R. Ali and M. M. Sufyan Beg. An overview of Web search evaluation methods. *Computers & Electrical Engineering*, 37(6):835–848, 2011. ISSN 0045-7906. doi: http://dx.doi.org/10.1016/j.compeleceng.2011.10.005.

[9] J. Bar-Ilan, M. Mat-Hassan, and M. Levene. Methods for comparing rankings of search engine results. *Computer Networks*, 50(10):1448–1463, 2006. ISSN 1389-1286. doi: http://dx.doi.org/10.1016/j.comnet.2005.10.020.

[10] Z.Q. Zhou, T.H. Tse, F.-C. Kuo, and T.Y. Chen. Automated functional testing of Web search engines in the absence of an oracle. Technical Report TR-2007-06, The University of Hong Kong Department of Computer Science, 2007. URL `http://www.csis.hku.hk/research/techreps/document/TR-2007-06.pdf`.

[11] T.Y Chen, T.H. Tse, and Z. Q. Zhou. Fault-based testing in the absence of an oracle. In *Proceedings of the 25th Annual International Computer Software and Applications Conference (COMPSAC 2001)*, pages 172–178, 2001. doi: 10.1109/CMPSAC.2001.960614.

[12] T. Imielinski and A. Signorini. If you ask nicely, I will answer: Semantic search and today's search engines. In *IEEE International Conference on Semantic Computing (ICSC '09)*, pages 184–191, September 2009. doi: 10.1109/ICSC.2009.31.

[13] T. Y. Chen, F.-C. Kuo, T. H. Tse, and Z. Q. Zhou. Metamorphic testing and beyond. In *Proceedings of the 11th Annual International Workshop on Software Technology and Engineering Practice*, pages 94–100, September 2003. doi: 10.1109/STEP.2003.18.

[14] Z. Q. Zhou, D. H. Huang, T. H. Tse, Z. Yang, H. Huang, and T. Y. Chen. Metamorphic testing and its applications. In *Proceedings of the 8th International Symposium on Future Software Technology (ISFST 2004)*. Software Engineers Association, 2004.

[15] B. Hailpern and P. Santhanam. Software debugging, testing, and verification. *IBM Systems Journal*, 41(1):4–12, January 2002. ISSN 0018-8670. doi: 10.1147/sj.411.0004.

[16] L. J. Morell. A theory of fault-based testing. *IEEE Transactions on Software Engineering*, 16(8):844–857, August 1990. ISSN 0098-5589. doi: 10.1109/32.57623.

[17] T. Y. Chen, T. H. Tse, and Z. Q. Zhou. Fault-based testing without the need of oracles. *Information and Software Technology*, 45(1):1–9, 2003. ISSN 0950-5849. doi: http://dx.doi.org/10.1016/S0950-5849(02)00129-5. URL `http://www.sciencedirect.com/science/article/pii/S0950584902001295`.

[18] H. Liu, X. Liu, and T. Y. Chen. A new method for constructing metamorphic relations. In *Proceedings of the 12th International Conference on Quality Software*, pages 59–68, August 2012. doi: 10.1109/QSIC.2012.10.

[19] M. Asrafi, H. Liu, and F.-C. Kuo. On testing effectiveness of metamorphic relations: A case study. In *Proceedings of the 5th International Conference on Secure Software Integration and Reliability Improvement*, pages 147–156, 2011. doi: 10.1109/SSIRI.2011.21.

[20] T. Y. Chen, J. W. K. Ho, H. Liu, and X. Xie. An innovative approach for testing bioinformatics programs using metamorphic testing. *BMC Bioinformatics*, 10: 10–24, January 2009.

[21] P. Wu. Iterative metamorphic testing. In *Proceedings of the 29th Annual International Computer Software and Applications Conference (COMPSAC 2005)*, volume 2, pages 19–24, July 2005.

[22] H. Liu, F.-C. Kuo, D. Towey, and T. Y. Chen. How effectively does metamorphic testing alleviate the oracle problem? *IEEE Transactions on Software Engineering*, 40(1):4–22, Jan. 2014. ISSN 0098-5589. doi: 10.1109/TSE.2013.46.

[23] Y. Cao, Z. Q. Zhou, and T. Y. Chen. On the correlation between the effectiveness of metamorphic relations and dissimilarities of test case executions. In *Proceedings of the 13th International Conference on Quality Software (QSIC 2013)*, pages 153–162. IEEE Computer Society Press, 2013.

[24] J. Chen, F.-C. Kuo, X. Xie, and L. Wang. A cost-driven approach for metamorphic testing. *Journal of Software (1796217X)*, 9(9):2267 – 2275, 2014. ISSN 1796217X. URL `http://search.ebscohost.com.ezproxy.uow.edu.au/login.aspx?direct=true&db=iih&AN=98496935&site=ehost-live`.

[25] U. Kanewala. Techniques for automatic detection of metamorphic relations. In *2014 IEEE Seventh International Conference on Software Testing, Verification and Validation Workshops (ICSTW)*, pages 237–238, March 2014. doi: 10.1109/ICSTW.2014.62.

[26] M. S. Sadi, F.-C. Kuo, J. W. K. Ho, M. A. Charleston, and T. Y. Chen. Verification of phylogenetic inference programs using metamorphic testing. *Journal of Bioinformatics and Computational Biology*, 9(6):729–747, December 2011.

[27] X. Xie, J. W. K. Ho, C. Murphy, G. Kaiser, B. Xu, and T. Y. Chen. Testing and validating machine learning classifiers by metamorphic testing. *Systems and Software*, 84(4):544–558, April 2011.

[28] Y. Yao, C. Zheng, S. Huang, and Z. Ren. Research on metamorphic testing: A case study in integer bugs detection. In *2013 Fourth International Conference on Intelligent Systems Design and Engineering Applications*, pages 488–493, Nov 2013. doi: 10.1109/ISDEA.2013.516.

[29] J. Mayer and R. Guderlei. On random testing of image processing applications. In *Proceedings of the 6th International Conference on Quality Software (QSIC 2006)*, pages 85–92, October 2006. doi: 10.1109/QSIC.2006.45.

[30] K. Y. Sim, D. M. L. Wong, and T. Y. Hii. Evaluating the effectiveness of metamorphic testing on edge detection programs. *International Journal of Innovation, Management and Technology*, 4(1):6–10, February 2013. Copyright - Copyright IACSIT Press Feb 2013; Last updated - 2013-12-18.

[31] W. K. Chan, T. Y. Chen, H. Lu, T. H. Tse, and S. S. Yau. A metamorphic approach to integration testing of context-sensitive middleware-based applications. In *Proceedings of the 5th International Conference on Quality Software (QSIC 2005)*, pages 241–249, 2005. doi: 10.1109/QSIC.2005.3.

[32] S. Segura, R. M. Hierons, D. Benavides, and A. Ruiz-Cortés. Automated metamorphic testing on the analyses of feature models. *Information and Software Technology*, 53(3):245–258, 2011. ISSN 0950-5849. doi: http://dx.doi.org/10.1016/j.infsof.2010.11.002.

[33] J. Mayer and R. Guderlei. An empirical study on the selection of good metamorphic relations. In *Proceedings of the 30th Annual International Computer Software and Applications Conference (COMPSAC'06)*, pages 475–484, September 2006. doi: 10.1109/COMPSAC.2006.24.

[34] D. Hawking, N. Craswell, P. Thistlewaite, and D. Harman. Results and challenges

in Web search evaluation. *Computer Networks*, 31(11-16):1321–1330, 1999. ISSN 1389-1286. doi: http://dx.doi.org/10.1016/S1389-1286(99)00024-9.

[35] S. J. Clarke and P. Willett. Estimating the recall performance of Web search engines. In *Aslib Proceedings*, pages 184–189. MCB UP Ltd, 1997.

[36] L. T. Su. A comprehensive and systematic model of user evaluation of Web search engines: II. An evaluation by undergraduates. *Journal of the American Society for Information Science and Technology*, 54(13):1193–1223, November 2003. ISSN 1532-2890. doi: 10.1002/asi.10334.

[37] J. Bar-Ilan, K. Keenoy, E. Yaari, and M. Levene. User rankings of search engine results. *Journal of the American Society for Information Science and Technology*, 58(9):1254–1266, July 2007. ISSN 1532-2890. doi: 10.1002/asi.20608.

[38] S. Lawrence and C. L. Giles. Searching the World Wide Web. *Science*, 280:98–100, 1998.

[39] L. Vaughan and M. Thelwall. Search engine coverage bias: Evidence and possible causes. *Information Processing & Management*, 40(4):693–707, July 2004. ISSN 0306-4573. doi: http://dx.doi.org/10.1016/S0306-4573(03)00063-3.

[40] L. Zhao. Jump higher: Analyzing web-site rank in Google. *Information Technology and Libraries*, 23(3):108–118, September 2004.

[41] L. Vaughan. New measurements for search engine evaluation proposed and tested. *Information Processing & Management*, 40(4):677–691, July 2004. ISSN 0306-4573. doi: http://dx.doi.org/10.1016/S0306-4573(03)00043-8.

[42] I. Soboroff, C. Nicholas, and P. Cahan. Ranking retrieval systems without relevance judgments. In *Proceedings of the 24th Annual International ACM SIGIR*

Conference on Research and Development in Information Retrieval (SIGIR'01), pages 66–73, New York, NY, 2001. ACM Press.

[43] J. A. Aslam and R. Savell. On the effectiveness of evaluating retrieval systems in the absence of relevance judgments. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval (SIGIR'03)*, pages 361–362, New York, NY, 2003. ACM Press.

[44] F. Can, R. Nuray, and A. B. Sevdik. Automatic performance evaluation of Web search engines. *Information Processing & Management*, 40(3):495–514, 2004. ISSN 0306-4573. doi: http://dx.doi.org/10.1016/S0306-4573(03)00040-2.

[45] W. Zheng, H. Ma, M. R. Lyu, T. Xie, and I. King. Mining test oracles of Web search engines. In *Proceedings of the 26th IEEE/ACM International Conference on Automated Software Engineering*, pages 408–411, November 2011.

[46] C. Murphy, K. Shen, and G. Kaiser. Using JML runtime assertion checking to automate metamorphic testing in applications without test oracles. In *Proceedings of the International Conference on Software Testing Verification and Validation (ICST'09)*, pages 436–445, April 2009.

[47] T. Y. Chen, F.-C. Kuo, and Z. Q. Zhou. An effective testing method for end-user programmers. In *Proceedings of the First Workshop on End-user Software Engineering (WEUSE I)*, pages 1–5, New York, NY, May 2005. ACM Press. doi: 10.1145/1082983.1083236.

[48] I. S. Altingovde, R. Blanco, and B. B. Cambazoglu. Characterizing Web search queries that match very few or no results. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management (CIKM'12)*, pages 2000–2004, New York, NY, 2012. ACM Press.

[49] H. Long, B. Lv, T. Zhao, and Y. Liu. Evaluate and compare Chinese internet search engines based on users' experience. In *Proceedings of the International Conference on Wireless Communications, Networking and Mobile Computing (WiCom 2007)*, pages 6134–6137, September 2007. doi: 10.1109/WICOM.2007.1504.

[50] B. J. Jansen. The comparative effectiveness of sponsored and nonsponsored links for Web e-commerce queries. *ACM Transactions on the Web*, 1(3), May 2007.

[51] English word dictionary. `http://docs.oracle.com/javase/tutorial/collections/interfaces/examples/dictionary.txt`.

[52] Google. Search operators. `https://support.google.com/websearch/answer/136861?hl=en&ref_topic=3180167`.

[53] Bing. Search operators. `http://onlinehelp.microsoft.com/en-us/bing/ff808421.aspx`, .

[54] Bing Chinese. Search operators. `http://onlinehelp.microsoft.com/zh-cn/bing/ff808421.aspx`.

[55] Baidu. Search operators. `http://help.baidu.com/question?prod_en=webmaster&class=123&id=505`.

[56] Bing. Advanced search options. `http://onlinehelp.microsoft.com/en-us/bing/ff808438.aspx`, .

[57] H. Hou, H. Li, and J. Wen. A comparative study for search engines business model - based on the case of Baidu and Google. In *Proceedings of the 2010 International Conference on E-Business and E-Government*, pages 228–232, May 2010. doi: 10.1109/ICEE.2010.65.