6-17-2019 2:00 PM

How to Rank Answers in Text Mining

Guandong Zhang, *The University of Western Ontario*

Supervisor: Hao Yu, *The University of Western Ontario*

A thesis submitted in partial fulfillment of the requirements for the Doctor of Philosophy degree in Statistics and Actuarial Sciences

Abstract

In this thesis, we mainly focus on case studies of user submitted answers. We assess the performance and ranking quality of CEW-DTW and improve upon it by combining it with Kullback-Leibler divergence.

Because the CEW-DTW and KL-CEW-DTW only consider frequency of keywords and noise rather than the probability distributions, we are able to improve upon them by introducing a measure known as General Entropy. Using this new measure, we attempt to find an objective goal – called the Maximum General Entropy - which can be regarded as a standard by which to assess user answers. Each answer can be compared against this value to determine the quality of that answer with regards to probabilities of keywords and "noise" words. This methodology is applied to a corpus of answers to Amazon questions.

We further develop this methodology to assess inner connections among keywords and noise. The concept of General Entropy is extended to Transition Probability Entropy, which assesses the probability of transitioning from one word to another in comparison to all other possible transitions. This method also leads to a measure of the information contained within an answer.

Finally, we show the process of cleaning the Amazon data for the above analyses and use the larger corpus to assess the strengths and weaknesses of each of the methodologies and whether simpler methodologies can be used in place of the more complex ones.

Keywords

CEW-DTW, KL-CEW-DTW, Demotion, Promotion, Keywords, Noise, the General Entropy, the Maximum General Entropy, Markov Transition Matrix, the Total Transition Probability Entropy

Co-Authorship Statement

This work was completed under the supervision of Dr. Hao Yu. All papers about KL-CEW-DTW, the General Entropy, and the Markov Transition Probability Entropy will be coauthored with Dr. Hao Yu, and some papers with Dr. Lu Xiao.

Paper title: CEW-DTW: A new time series model for text mining

Publication: In 2018 International Conference on Information and Communications Technology (ICOIACT), pp. 158-162. IEEE

List of authors: GuanDong Zhang, Hao Yu, and Lu Xiao

The ideal to analyze text ranking is inspired by data of oral history, which was introduced in Centre for Oral History and Digital Storytelling (COHDS) in Concordia University. However, these data are not public. So, we find a public data source for our analysis.

Acknowledgments

During the progress of my research, I cannot complete all researches and the dissertation without the assistance and help from other people. I would like to pay homage to my supervisor, Dr. Hao Yu, to guide my Ph.D. research. His abiding patience, academic passion as well as abundant knowledge support me in the progress of my research. In most of the time, his excellent suggestion always enables me to open my mind.

Also, I would like to take the chance to thank all examiners: Professor Ying Zhang, Professor Xin Wang, Professor Ian McLeod, and Professor Douglas Woolford. They spend much time to read my thesis as well as achieve my defense. I also appreciate some, but not limited to, professors: Dr. Ian McLeod, Dr. Ricardas Zitikis, Dr. Reg Kulperger in Western University and Dr. Lu Xiao in Syracuse University for their advices in the progress of my research. I also thank Professor Steven High and Centre for Oral History and Digital Storytelling (COHDS) in Concordia University. Though I did not use their data finally, their suggestions help me a lot in my research.

Additionally, I would like to spread my heartfelt gratitude to my parents. Though they are not in Canada, they really help me a lot to look after my daughter with their greatest patience. Especially, my parents try their best to support me in finance when I am in trouble. I also thank my wife, Minjie Huang, who stay with my parents to take care of my daughter in China. My relatives unselfishly give their comprehension, toleration and love to me. I also thank my lovely daughter Yifei Zhang, who is a sensible clever girl and enables me to complete Ph.D. confidently.

Last but not least, I would like to thank all my friends and colleagues, Chen Yang, Dazhong (Dexen) Xi, Devan Becker, John Thompson, TianPei Jiang, WenJun Jiang, YunHao Lai, Ning

Sun, WenKai Ma, Ning Sun, Kexin Luo and so on. They take a very important role in the progress of my research at Western University.

Table of Contents

List of Tables

List of Figures

Chapter 1

1 Research Background

In our world, information is delivered and received through various data forms such as sound tracks, video clips, texts, and so on. Among all the data forms, texts are traditional but efficient way to deliver information accurately. As a result, a huge amount of texts usually contains crucial information. For example, comments on Amazon, Ebay or TripAdviser are important references for consumers' or tourists' decision-making and therefore are actually crucial for the websites to study the consumers' or tourists' behavior. In response to these demands, data analysis methods focusing on texts become more and more popular in the modern world.

Currently, many approaches for analyzing texts have been developed to extract information mainly by classification, clustering and ranking. Though these research approaches analyze texts from different viewpoints, the main purpose of these approaches is to extract available information so as to make readers understand texts efficiently. Text mining mainly includes four research field: Text Classification, Text Clustering, Text Pattern Recognition, and Text Ranking. Different contents or topics are interested in these fields.

1.1 Text Classification

The most usual application of text classification is filtering spam for emails. As people do not want to waste their time on reading less important emails, spam filtering techniques could help improve their working efficiency. The major concern to spam filtering techniques is how to avoid identifying true important emails as spams. In general, the classification error is always the most important measure of performance for text classification methods. However, computational efficiency is also important when developing an application in real-world scenarios.

Many approaches have been developed to do text classification. Formally, one text can be considered as a document $\boldsymbol{K_i}$, which belongs to a part of a set of documents **Q**. Since we have a category pool $\{C_1, C_2, C_3, \dots, C_n\}$, the purpose of the text classification is to put a category tag to this document (Ikonomakis et al. [1]). Onan and Koruko$\breve{g}$lu [2] develop an aggregation-based feature selection model to extract key information from documents for classification. This methodology applies K-nearest or Naive Bayes to be training models and has been proven to have a higher accuracy than other individual methods. Liu et al. [3] develop a multi-task learning framework. This model mitigates latent features to be the public or private pool to conflict each other. This model applies long short-term memory and has been proven to be helpful to several text classification tasks. Xuan et al. [4] explore a semi-supervised text classification approach to classify bugs to improve the bug report quality. By combining Naive Bayes classifier and the expectation-maximization together, this model can handle different kinds of bug reports and show a high classification accuracy. Xu [5] combines the Naive Bayesian model with Multinomial, Bernoulli and Gaussian models respectively to be three new models. By comparing these three models, the author illustrates that the Naive Bayesian classifier with Bernoulli model shows an equal classification effect with the Bayesian counterpart. Chen et al. [6] use the symmetric KL-divergence to develop a new model to a new methodology, which can measure centroid in text classification. This model is based on the document distribution and the document centroid. It has been proven to have a better classification quality than the Naive Bayes methodology. Garg et al. [7] do research about counterfactual fairness. When they analyze texts, they focus on a special question to classify. They use three methodologies to analyze texts: Hard ablation, Blindness and Counterfactual Logit Pairing. These methodologies have been proven to increase the detection quality of counterfactual fairness. Shu et al. [8] develop a model, Deep Open Classification, to handle the open classification problem. This model is based on Convolutional Neural Network and has shown a better performance than the state-of-the-art methods. Based on Long Short-Term Memory, Yogatama et al. [9] analyze discriminative models and generative models respectively in error rates. They conclude that the generative model shows a better performance than the discriminative model when the data size is small. Ive et al. [10] mainly care about mental health problems. They adapt a hierarchical Recurrent Neural Network (RNN) to classify mental health posts. This model has shown a better performance than Convolutional Neural Network in terms of the F-measure assessment. Li and Ye [11] firstly analyze a framework called “Reinforcement

Learning based Adversarial Networks for Semi-supervised learning" (RLANS). This framework contains two parts: prediction and judgement, and they can be applied in the discrete data without data generation. They develop a semi-supervised model for text classification. This model is proven to perform better than some current semi-supervised models, such as LSTM, SeqSSL, SeqSSL+VAT and so on. Liu et al. [12] develop two classification models to analyze concept information: "the neural bag of words with direct mapping" (NBOW-DM) and "the neural bag of words with gated mapping" (NBOW-GM). These two models are based on the neural classification. The second model is proven to be better than the first one in performance. They are all proven to be less time consuming than counterparts. Saha et al. [13] analyze the wrong comment problem, including missing-item return comments, comment-mismatch and non-comment-mismatch. They develop a methodology about labeling functions, which focuses on useful information as well as noises. This methodology is proven to perform better than some Machine Learning and Deep Learning models, such as Xgboost, Xgboost+filtering, BLSTM, and BLSTM+noise-aware.

1.2 Text Clustering

Clustering is another important research field in data analysis, including text mining. The task of text clustering is to separate an original data to several groups according to certain features. Texts within a group should show no differences with respect to the selected features. Features are usually defined by similarity functions. Text clustering could improve querying speed through dividing text domains such as whole documents, paragraphs and even sentences into categories.

Currently, we can find many approaches to do text clustering. Abualigah et al. [14] apply the particle swarm optimization algorithm to develop a feature selection methodology. This algorithm is based on the term frequency-inverse document frequency. This model applies k-mean to find features for clustering. It has been proven to improve clustering efficiency and decrease the model implement time. Xu et al. [15] develop a methodology to find appropriate parameters when they use one text clustering algorithm. They analyze cognitive psychology to find basic documents

categories. This methodology is proven to perform better than some clustering methodologies, such as k-mean, single linkage clustering. A new hierarchical text clustering methodology, called as FireflyClust, is developed by Mohammed et al. [16]. This methodology is based on Cosine Similarity and relocates procedure to enhance clustering accuracy. It is proven to have a better performance than Bisect K-means, hybrid Bisect K-means and PGSCM. Grieco et al. [17] apply text clustering in natural language documents. They analyze industry process when Engineering change happens. Their model is based on TF-IDF. It uses Self Organizing Map to cluster Engineering Change Requests documents. Assessment results show that this methodology enhances the efficiency of reusing or exploiting knowledge. Xu et al. [18] focus on the research field of neural network. They develop a framework named Self-Taught Convolutional network to study short texts. This framework can be applied to enhance performance of four dimensionality reductions: Average Embedding, Latent Semantic Analysis, Laplacian Eigenmaps and Locality Preserving Indexing. Dörpinghaus et al. [19] explore a graph-theoretical approach to cluster documents. The approach, named as PS-Document Clustering, is developed from some similarity methods, such as Tanimoto similarity or TF-IDF. This methodology transfer documents' distances to graph distances in order to separate documents. It has been proven to be extraordinary in documents clustering. Matei et al. [20] use time series theory to cluster documents. The methodology is based on Dynamic Time Warping (DTW) and K-Medoids. It regards TF-IDF and Cosine Similarity as the baseline to cluster chapters of a document. This methodology has been proven to be an efficient on when it is tested in the writing of Lev Nikolaevici Tolstoy and Feodor Dostoevsky. Abualigah et al. [21] develop a model by applying feature weight scheme and dynamic dimension reduction to select features. Then they apply k-mean to cluster documents according to these features. They prove that this model performs better than some state-of-the-art methods, such as GVSM-SFS, GVSM-HFS, BPSO, PM, FW-PSO-DDR, etc. Abualigah and Khader. [22] explore a hybrid algorithm to cluster documents. This methodology is developed from the hybrid PSO algorithm with the GOs. Compared with K-mean clustering methodology, this methodology has a better performance.

1.3 Text Pattern Recognition

Text recognition can also be considered as text pattern recognition. It is to identify laws contained in texts having similar characteristics in terms of some algorithms. For example, as someone is acknowledged to author several books and articles, text recognition could help identify whether an article having similar writing style with unknown author is written by this one or not. In artificial intelligence, this technique could help robots to "chat" with real people by capturing patterns of chats under certain scenarios.

There are many literatures about text recognition research. Lu et al. [23] develop a new framework to extract texts from shadowed text images. They firstly transfer text images to binary images by applying a local adaptive threshold method. Then, they use a projection-based denoising method and a median filter method to remove noises to obtain clear image files. This framework is proven to show a good Optical Character Recognition accuracy for Tesseract drops. In terms of Bayesian theory, Tian et al. [24] develop a model to track, detect and recognize texts embedded in videos. They use Hungarian algorithm to calculate similarities between trajectories and detection objects so that they can track texts. This framework is proven to show a better performance when it is compared with other general models. Yang et al. [25] develop an adaptive ensemble of deep neural networks to recognize texts in a picture when this picture has a complicated background. This model is based on a Bayesian Model. Assessment results illustrate that AdaDNNs shows a 10% improvement in terms of the baseline DNNs. Shi et al. [26] explore a methodology named Convolutional Recurrent Neural Network to consider sequence in the image. This methodology is based on Deep Convolutional Neural Networks and Recurrent Neural Networks. According to assessment results, CRNN show a better performance than other two methodologies: Capella Scan and PhotoScore. Bušta et al. [27] develop a framework to locate and recognize text in images. They use the YOLOv2 architecture to improve the image recognition accuracy. Also, they find the Region Proposal Network is a good methodology to achieve region proposals. The bilinear sampling method are applied to generate the object map for feature representation. Assessment results show that this methodology has a better performance than other models in F-measure test. Xie et al. [28] develop a multi-spatial-context fully convolutional recurrent network to recognize Chinese handwritten online. This model analyzes signature path by applying spatial structure and

pen-tip trajectories information. It illustrates a better performance than some other models in Chinese handwritten detection. Liu et al. [29] develop a model named SqueezedText to detect real-time scene text. This model uses a binary convolutional encoder-decoder neural network and a backend bidirectional recurrent neural network to deal with text. It demonstrates a good enhancement in run-time speed, memory usage and accuracy. Liao et al. [30] develop an end-to-end trainable fast scene text methodology to detect text. This is an end-to-end fully convolutional network and based on the loss function. It improves the text location speed in images. Compared with other models, this model enhances the recognition accuracy as well as the implement speed.

1.4 Text Ranking

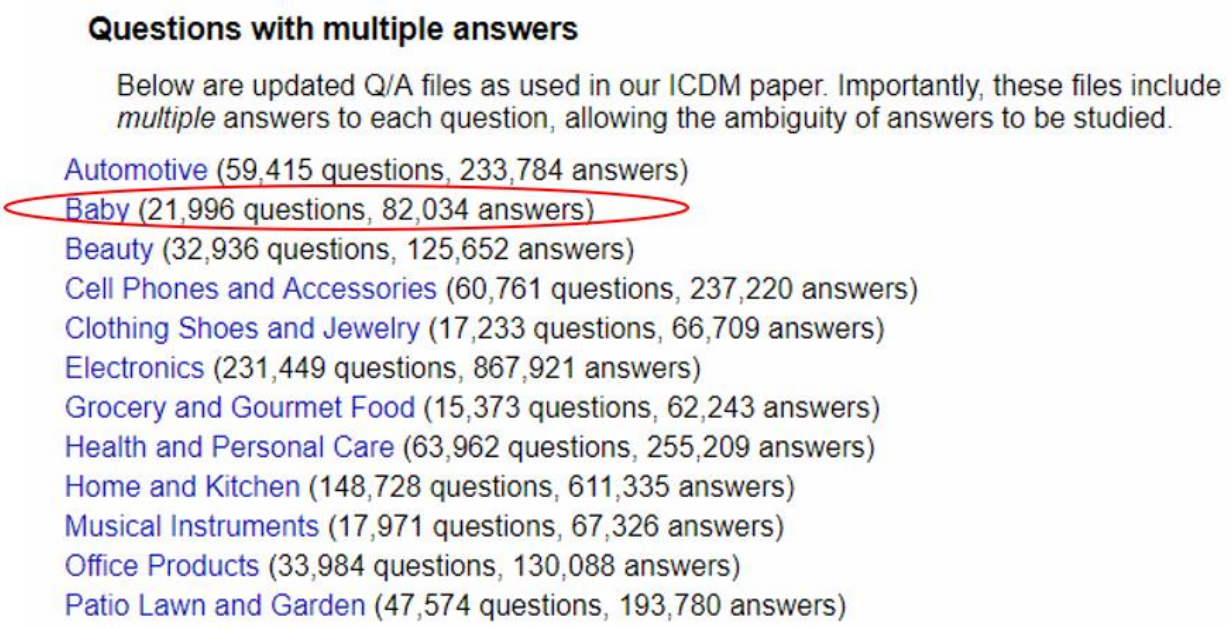
Text ranking is the basic task for many important applications such as developing search engines. The task aims to rearrange objects such that objects of higher qualities could be found more easily according to certain rules. For search engines, information that is more relevant to the keywords should be assigned higher ranks. And the idea is the same for other text ranking tasks.

Many literatures have been published to discuss text ranking. Raifer et al. [31] take authors' action for analysis in order to improve the ranking quality of these authors' documents. They use theoretical methods and empirical methods to do their research. In theoretical methods, "repeated game" and "minmax regret equilibrium" are applied to uphold goodness of publications. In empirical methods, they try to make current documents ranking be similar to the previous ranking. These methods are proven to demonstrate a high accuracy in documents ranking. Xiong et al. [32] combine the query entity linking method and the entity-based document ranking method together. They develop a joint model, which is called as JointSem. This model firstly makes three actions: (1) to spot n-grams query in a dictionary; (2) to link entities with spotted surfaces; (3) to rank linked entities. Then, this model generates an objective function about these three actions to be a ranking function, which is applied to verify the document ranking quality. Assessment results illustrate that JointSem perform better than other models, such as RankSVM. Pandey et al. [33] explore a Linear feature extraction algorithm to rank documents. In their research, each document

can be regarded as a matrix. The key step is to transfer an original matrix to be a low-dimension matrix by decreasing the dimension of document vectors. This model uses linear approach to extract key information. This model is proven to perform better than GAS, FSMSVM and LifeRank in terms of the normalized discounted cumulative gain (nDCG) evaluation rule. Wang et al. [34] explore a graph-based methodology to rank documents. They use Topical Tripartite Graph model to explore a ranking methodology. This model applies a random walk algorithm to test distances of entities so as to find a good ranking. Based on Markov theory, Wei et al. [35] develop a rank model named MDPRank. This model combines Monte-Carlo Stochastic algorithm in the information retrieval method. By applying nDCG assessment, this model performs better than other models, such as RankSVM, ListNet, AdaRank-MAP and so on. Xiong et al. [36] use the ad-hoc retrieval method to develop an attention-based ranking model AttR-Duet. This model lowers noise parts and apply the word-entity duet to rank texts. This model is based on the Convolutional Neural Network. This model performs remarkable in TagMe Accuracy as well as Attention Gain. Fang et al. [37] explore a word-sentence co-ranking model named CoRank to obtain documents' summarization automatically. This model analyzes the correlation of word-sentence and connects this correlation with the graph-based ranking model. Words and sentences are assigned with different weights for analysis in this model. A redundancy elimination technique is also applied in this model.

1.5 Data Description

Before we start comparison of methodologies, we introduce how to obtain data for our analysis. We use an open dataset: Amazon data (http://jmcauley.ucsd.edu/data/amazon/qa/). These data are constructed by Question and Answer (Wan and Julian [38], McAuley and Alex [39]). It is from Amazon. The total data volume is approximately 1.4 million questions, which have been answered. According to the description, this data includes Amazon product review data and is constructed by matching ASINs in the Q/A dataset. The review also contains product metadata (product titles etc.). We choose answers of "Baby" category in Amazon data as examples.



Questions with multiple answers

Below are updated Q/A files as used in our ICDM paper. Importantly, these files include *multiple* answers to each question, allowing the ambiguity of answers to be studied.

Automotive (59,415 questions, 233,784 answers)
Baby (21,996 questions, 82,034 answers)
Beauty (32,936 questions, 125,652 answers)
Cell Phones and Accessories (60,761 questions, 237,220 answers)
Clothing Shoes and Jewelry (17,233 questions, 66,709 answers)
Electronics (231,449 questions, 867,921 answers)
Grocery and Gourmet Food (15,373 questions, 62,243 answers)
Health and Personal Care (63,962 questions, 255,209 answers)
Home and Kitchen (148,728 questions, 611,335 answers)
Musical Instruments (17,971 questions, 67,326 answers)
Office Products (33,984 questions, 130,088 answers)
Patio Lawn and Garden (47,574 questions, 193,780 answers)

When we obtain answer examples, we would like to clean and ontain original answers for analysis (see Chapter 6 about how to deal with data).

1.6 Objects and Results

This research is a cross-disciplinary research between statistics and Artificial Intelligence. Since artificial intelligence is to fit data in machine learning, we try to use our statistical methodologies to explain data about what is going on from the viewpoint of artificial intelligence.

In this thesis, we mainly analyze contents of answers. We present the methodology CEW-DTW and assess its performance about ranking quality in Chapter 2. Since we can regard a sentence as a time series sequence, we develop CEW-DTW in terms of a time series methodology: Dynamic Time Warping. When we want to assess the ranking quality of a group of answers, we design an "ideal" answer as a standard to rank answers. We use the normalized discounted cumulative gain to test the performance of CEW-DTW. This criterion illustrates that the performance of CEW-DTW is better than previous methodologies, such as Dynamic Time Warping and Dynamic Time Warping-Delta. Based on the CEW-DTW, we improve this methodology by combining Kullback-

Leibler divergence with CEW-DTW in Chapter 3, since Kullback-Leibler divergence can check the difference of probability distributions in two sequences. The new methodology KL-CEW-DTW is proven to perform better than CEW-DTW in ranking according to the criterion of the normalized discounted cumulative gain . However, CEW-DTW and KL-CEW-DTW assess answers in terms of the distance to an "ideal" answer. They do not analyze answers from the viewpoint of probability. Therefore, in Chapter 4, we introduce a new methodology, the General Entropy, to see how probabilities of noise and keywords affect qualities of answers. We mainly give some properties of the general entropy. We firstly analyze the value range of the General Entropy in different noise probability conditions. Also, we illustrate that the value of the general entropy is always equal to 0, if the length of an answer is 1 (**Note**: the length of an answer represents the number of words in this answer). From the view point of uniform distribution, we give the definition of the global entropy, which can be applied to prevent fake answers. Since the assessment of CEW-DTW and KL-CEW-DTW is based on the distance to an "ideal" answer, we try to find an objective goal so as to judge actual answers with respect to this goal. Therefore, we introduce the maximum general entropy. We try to use the general entropy methodology to find an imaginary answer with the maximum general entropy from the mathematical viewpoint (though this answer may not exist). This answer can also be regarded as an "ideal" answer. Here, we give definitions of demotion and promotion about keywords. Thus, we can use demotion and promotion to assess keywords in terms of the maximum general entropy answer. Then, we analyze the value range of the global probability of noise. In such situation, the maximum general entropy probability of noise is smaller than the global probability of noise. According to the range of the global probability of the keyword, we analyze how the keyword is promoted or demoted. Here, we find two value: Q_L and Q_H. We find that the keyword is promoted when the global probability of the keyword is between Q_L and Q_H. Otherwise, the keyword is demoted. Then, we give the definition about how to determine the optimum number of keywords. However, the optimum number of keywords is usually smaller than the original number of keywords. So, we show the formula of relative efficiency in terms of different numbers of selected keywords. In order to assess the general entropy, we simulate some global probabilities and maximum general entropy answers for comparison. We also adapt Amazon data to assess these presented formulas. Additionally, we compare global probabilities and maximum general entropy answers to find their relationships. We also apply these two kinds of probabilities in Amazon data to see how many keywords are

enough for analysis. Additionally, we choose some answers with high or low CEW-DTW values to see how probabilities of these answers are consistent with their maximum entropy probabilities. Comparison results illustrate that the Low-CEW-DTW answer has the lower probability of noise and higher probabilities of keywords than those of the High-CEW-DTW answer respectively. Also, we find that the global entropy is between the general entropy of High-CEW-DTW answer and the general entropy of Low-CEW-DTW answer. We also organize a small group of survey to assess the general entropy. We also use comments of a real Amazon product to test the general entropy, because we want to see whether we can apply this methodology in industry. Survey results show that the General Entropy test is more reasonable than CEW-DTW. Though these developed methodologies can analyze answer qualities, they do not consider the inner connections among keywords and noise. In Chapter 5, we introduce the Markov Entropy in terms of the Markov transition matrix. We firstly get transition probabilities of noise and keywords. We approach another new entropy, the Transition Probability Entropy. We imitate propositions in Chapter 4 to present similar propositions. Meanwhile, we still adapt Amazon dataset to compare maximum transition entropy probabilities and global transition probabilities of noise and keywords respectively. Also, we find two value: Q_{ML} and Q_{MH}, which can be used to see whether the transition of two words is promoted or demoted. Similarly, we also use the same real Amazon product to see whether we can apply this methodology in industry. In Chapter 6, we illustrate how to obtain original answers. Then, we present how to remove stopping words and collinearity to get answers for analysis. We compare our developed methodologies to see how these methodologies are consistent. We also introduce Wald–Wolfowitz runs test and compare it with developed methodologies to verify their relationships. Finally, we get conclusions about consistence of these methodologies. In Chapter 7, we introduce some future research plans to extend our methodologies.

Chapter 2

2 CEW-DTW: A new time series model for text mining

The keyword information is usually applied to describe answers. In most of the previous studies, researchers usually rank answers according to keyword retrieval, which fails to consider the importance of the time sequence of keywords in answers. In this chapter, we propose CEW-DTW, a new time series model for answer ranking. This model considers the importance of the time sequence of keywords as well as the number of keywords. CEW-DTW is developed from a carefully designed model, Dynamic Time Warping-Delta (DTW-D). We choose Amazon question/answer data as our evaluation dataset. We apply Entropy to remove redundant noise in answer vectors. In experiments, we apply normalized discounted cumulative gain (nDCG) as the assess rule to test models. CEW-DTW is proven to have a better performance than Dynamic Time Warping (DTW) and Dynamic Time Warping-Delta (DTW-D) in answer ranking. An extensive set of evaluation results demonstrates the effectiveness of the CEW-DTW model for answer ranking.

2.1 Introduction

Question-answering (QA) has acted as an important role in many research fields, such as advanced web search (Etzioni [40], Sun et al. [41]). Instead of reading all answers, the users can save time by reading those relevant answers directly. Therefore, it becomes an important task for researchers to find the most relevant answers. Since each answer is combined with text and completed in a set time, we can view an answer as a time sequence. Many researchers have applied the time series to analyze answers (O'Connor et al. [42], Ishikawa [43]).

In this chapter, we develop a novel methodology to rank answers. To pursue this work, we base our work on the public data. Amazon question/answer data is a kind of famous public data since it has been applied in opinion-question answering systems research as well as developed in queries about customer reviews (see Chapter 6 about details of data).

This chapter describes some new results by analyzing the answer data. We provide a novel time series algorithm to rank answers explicitly. Given the information that is provided by Amazon data, our model ranks answers by calculating the dynamic time warping between a given answer and the ideal answer. In the experimental evaluation, we illustrate that the quality of this novel rank is better than other chosen rank algorithms.

We initially choose interview data of oral history from Centre for Oral History and Digital Storytelling (COHDS) for analysis. However, this data is private and cannot be reviewed by other researchers at that moment. Therefore, we choose an open data for our analysis (see Chapter 6 about details of data)

2.2 Literature Review

Answer ranking has been regarded as an important assignment when we want to conduct research in the field of information retrieval. Keywords are important factors in ranking research since the extraction of keywords is currently considered an important application in many fields, such as document topics (Ventura and Silva [44]). These words can be considered as key information to describe documents. They illustrate that one can easily understand which documents can be read and which cannot. Jurczyk and Agichtein [45] have tested user expertise by measuring link analysis of answer graphs. They assume that answers, which are provided by authoritative users, have high qualities. Zhou et al. [46] explore three kinds of user profile information for answer ranking in Community-Based Question Answering. Jeon et al. [47] have anticipated the answer quality by using non-textual features of the answers. Tu et al. [48] develop a method to find the similarity between the set of best answers and their questions.

Yu et al. [49] illustrate that most previous studies in this area adapt IR-style ranking, which fails to consider the importance of the query answers. Therefore, if we only apply keywords to rank

text, we only obtain the information those specific keywords reveal, and we cannot determine the relationship of one text to another. Thus, future research really needs to find novel methodology to rank text in terms of not only the amount of keywords but also the context. Since different texts are written or spoken in different durations, we find that Dynamic Time Warping (DTW) is an appropriate way to measure the relationship between the different time lengths of texts. Therefore, our methodology will be based on DTW.

2.3 Model Introduction

2.3.1 Dynamic Time Warping

In this thesis, the work is based on the time series model, Dynamic Time Warping. After Sakoe and Chiba [50] preliminarily introduced the idea of Dynamic Time Warping, DTW is a widely used algorithm for similarity measurement (Berndt and James [51]). Müller [52] applies DTW to find the most favorable alignment between two dependent time series vectors. Since DTW is applied in two sequences, which may be different in rate of change, we can regard this method to be a dynamic calculation to some extent. The original DTW is designed to compare two time sequences so as to find the warping between them. Tsinaslanidis et al. [53] find that the advantage of DTW is to measure two series vectors when they have different dimensions. The smaller the DTW value of two vectors, the greater similarity these vectors represent. The following formula illustrates how DTW works. There are two time sequences: $\boldsymbol{A} \coloneqq \{a_1, a_2, \dots, a_n\}$ and $\boldsymbol{B} \coloneqq \{b_1, b_2, \dots, b_m\}$, where $n > 0$ and $m > 0$. When a warping set, $\boldsymbol{W} \coloneqq \{w_1, w_2, \dots, w_H\}$, is designed to map $\boldsymbol{A}$ and $\boldsymbol{B}$, this set should satisfy the following conditions:

- $maximum\{m, n\} \leq H \leq n + m - 1$;
- Boundary Condition: $w_1 = (1,1)\ and\ w_H = (m, n)$;
- Continuity and Monotony: Suppose $w_{k-1} = (c', d')$ and $w_k = (c, d)$, then $0 \leq c - c' \leq 1$ and $0 \leq d - d' \leq 1$.

Therefore, an optimal warping path can be obtained by dynamic calculating as following:

$$\zeta(r, s) = d(a_r, b_s) + \min\{\zeta(r-1, s), \zeta(r-1, s-1), \zeta(r, s-1)\}, \qquad (2-1)$$

where, $r = 1,2, \dots, n$ and $s = 1,2, \dots, m$. According to the description of Bautista et al. [54], if we have two feature vectors about sequences a_r and b_s, $d(a_r, b_s)$ can be considered as a distance of them. Since we are only interested in finding the final warping path, we can use (2-1) to calculate DTW as following:

$$\boldsymbol{DTW}(A,B) = \min\left\{\frac{1}{H}\sqrt{\sum_{k=1}^{H}\zeta(w_k)}\right\}. \qquad (2-2)$$

2.3.2 Dynamic Time Warping-Delta

Though DTW can compare two time series vectors with different dimensions and calculate the distance of two vectors, Chen et al. [55] find that DTW cannot reflect the metafeature. They introduce another model called Dynamic Time Warping-Delta (DTW-D), which is developed from DTW by combining DTW with Euclidean Distance. Since both DTW and DTW-D can be applied to deal with unlabeled data by learning from partial labelled data, we say that DTW and DTW-D are all in accordance with a semi-supervised learning framework. By applying (2-2), the formula of DTW-D is as following:

$$\mathbf{DTW} - \mathbf{D}(X,Y) = \frac{\boldsymbol{DTW}(X,Y)}{\boldsymbol{ED}(X,Y) + \varepsilon}, \qquad (2-3)$$

where ε is positive, and its value is very small so as to avoid the denominator to be zero (Chen et al. [53]). $\boldsymbol{ED}(X,Y)$ is Euclidean Distance, it is defined in this way: Suppose two vectors with the same length, $\boldsymbol{U} \coloneqq \{u_1, u_2, \dots, u_n\}$ and $\boldsymbol{V} \coloneqq \{v_1, v_2, \dots, v_n\}$, then the distance from $\boldsymbol{U}$ to $\boldsymbol{V}$ is:

$$\boldsymbol{ED}(\boldsymbol{U},\boldsymbol{V}) = \sqrt{\sum_{i=1}^{n}(u_i - v_i)^2}\,.$$

In this chapter, if lengths of two vectors are not equal, we replenish zero at the end of the shorter vector and make it to be equal to the longer vector.

2.4 A New Time Series Model

2.4.1 Data Preparation

Since keywords are applied in the analysis of this chapter, when a keyword is selected, we will find its synonyms in (http://www.thesaurus.com/). Then, we combine this keyword and its synonyms to be a one-keyword group. In our research, we can not only choose one keyword to form a one-keyword group but also several keywords to form a several-keyword group. Here, we uniformly call them the keywords group. We match each word of a selected answer with the keywords' group. If one word is matched with the keywords' group, it will be assigned to be value 1; otherwise, it is 0. Therefore, an answer will be transferred to be a zero/one vector. Here, we use 1 and 0 to represent the keyword and the noise respectively. Because we only care about useful information and the number of noise will not affect a zero/one vector tendency or shape, we will apply the entropy theory to reduce 0. Suppose the number of zero in the zero/one vector of an answer is n ($n \geq 2$). We obtain $m = the\ integer\ part\ of\ log(n)$, then we compress each m zero to be one zero. If the final rest zero number is less than m, they will be compressed to one zero.

2.4.2 An "ideal" answer

Russo [56] states that more information could help customers make decisions and it would not hurt them. Thus, we hope customers' answers contain keywords as many as possible. Blooma et al. [57] clearly illustrate that the length of an answer is important to judge whether an answer is a good answer. They also find that readers like long answers. Pande et al. [58] make it clear that long answers have a great number of details, which can help customers understand more information. Hambleton and Kanjee [59] find that examinees quickly selected the longest answer in a translation test, since they consider the longest answer to be the correct one. It illustrates that users have a great interest in the longest answer. Therefore, we design the length of the "ideal" answer to be the maximum answer length in the test group. If there are m documents, Luo et al. [60] suggest decomposing a document in a n-dimensional space for analysis. They also state that

if this answer contains all keywords, the ideal answer should appear at the position of $P_{ideal} = \underbrace{[1,1,\dots,1]}_{m}$. Here, m represents the dimensions of this answer. It also means Thus, we expect to find an ideal answer at this position. However, such an ideal answer usually does not exist in real answers. Therefore, when we receive a group of answers, we can generate an "ideal" answer vector in the following way: the length of this vector is the maximum length of the answer sentence in the test group; each element of this vector is 1. It means each word of this "ideal" answer is the keyword. Another reason for regarding a vector, whose elements are all 1, as an "ideal" vector is that this vector has no zero elements. It means that the "ideal" answer does not contain any noise. So, if we apply the entropy rule to remove noise information in an "ideal" vector, we will get the same vector as the original. Thus, we can choose such a vector as an "ideal" vector. Long [61] illustrates the quasi-standard concept, which requires users not to follow this standard absolutely. Since the "ideal" answer may not exist in real documents and actual answers may contain several keywords, we can regard this "ideal" answer as quasi-standard (Long [61]) and compare actual answers to this standard.

2.4.3 CEW-DTW Model

Though DTW-D can rank answers, this method still has some weaknesses. Since we find that the Euclidean Distance only compares two vectors from the viewpoint of value, it means DTW-D cannot reflect the angle of two vectors accurately. For example, there are three vectors: $\boldsymbol{A} := (2,1)$, $\boldsymbol{B} := (0,1)$, and $\boldsymbol{C} := (1,1)$. When we apply the Euclidean Distance formula, though we find $\boldsymbol{ED}(A,C)$ is equal to $\boldsymbol{ED}(B,C)$, these vectors reflect different trends since they have different values. Therefore, we should think about vector trends so as to find similar vectors. Though the Cosine Similarity can be applied to rank answers, it is not a first-rank option to rank time series sequences.

We improve DTW-D (as shown in (2-3)) by combining the Cosine Similarity method in the denominator. Since vector elements represent word frequencies, these elements are nonnegative. It means the Cosine Similarity is also nonnegative and its value is between 0 and 1. The Cosine

Similarity value of 1 illustrates two vectors with the same orientation. If the Cosine Similarity value of two vectors is equal to 0, these two vectors are at 90^0. The general formula of Cosine Euclidean Warping-Dynamic Time Warping (CEW-DTW) is as following:

$$\mathbf{CEW-DTW}(X,Y) = \frac{k^2 \boldsymbol{DTW}(X,Y)}{\boldsymbol{ED}(X,Y)\sqrt{\left(1-k^2\boldsymbol{CS}(X,Y)\right)^2+\omega}}, \qquad (2-4)$$

where we define k as the number of involved methods. Since we apply three methods in the new model: Cosine Similarity, Dynamic Time Warping and Euclidean Distance, we let $k = 3$. **CS**(X, Y) is the Cosine Similarity, the formula of cosine similarity is as following: Suppose two vectors, $\boldsymbol{A} \coloneqq (a_1, a_2, \dots, a_n)$ and $\boldsymbol{B} \coloneqq (b_1, b_2, \dots, b_n)$, then the Cosine Similarity (CS) of $\boldsymbol{A}$ and $\boldsymbol{B}$ is:

$$\boldsymbol{CS}(\boldsymbol{A},\boldsymbol{B}) = \frac{\sum_{i=1}^{n} a_i b_i}{\sqrt[2]{\sum_{i=1}^{n} a_i^2}\,\sqrt[2]{\sum_{i=1}^{n} b_i^2}}.$$

According to the description of (Chen et al. [55]), ω is an extremely small positive quantity used to avoid divide-by-zero error. In our experiments, we set ω = 0:000001. This formula can be applied to describe the relationship of two vectors from the viewpoint of time series. Because ***CS***(X, Y) (Cosine Similarity) or ***ED***(X, Y) (Euclidean Distance) is usually applied for two vectors with the same length, if the length of two vectors is different, we have to replenish zeros in the shorter vector so as to enable the length of these two vectors to be the same. In our research, since an answer is considered as a time series sequence, we replenish zeros at the end of the shorter vector. Because zeros are regarded as noises in this research, these replenished zeros will not remove useful information when we calculate the Cosine Similarity or the Euclidean Distance. For example, suppose two vectors: $A \coloneqq \{0,0,1,0,1\}$ and $\boldsymbol{B} \coloneqq \{0,1,0\}$, we generate the vector B to be a new one as $\boldsymbol{B} \coloneqq \{0,1,0,0,0\}$. Ye [62] has demonstrated that the cosine value is zero when two vectors are zero vectors. We define the cosine similarity of two zero vectors as zero.

Let $Y \coloneqq$ *the "ideal" answer*, (2-4) will be changed to the following:

$$\mathbf{CEW-DTW}(X_i, Y) = \frac{k^2 \boldsymbol{DTW}(X_i, Y)}{\boldsymbol{ED}(X_i, Y)\sqrt{\left(1 - k^2 \boldsymbol{CS}(X_i, Y)\right)^2 + \omega}}, \qquad (2-5)$$

where $X_i, i = 1, 2, 3, \dots, n$ represents individual answers.

2.5 Evaluation

2.5.1 Evaluation Standard

When we develop a new methodology (as shown in (2-5)), we usually want to assess this methodology so that we can check whether it is better or not than the previous ones. Wang et al. [63] and Baltrunas et al. [64] used the rank assessment rule –normalized Discounted Cumulative Gain. We adapt this assessment in this research. Let $p_1, p_2, \dots, p_k$ be a list of items. Let r_{p_i} be the true rating of the item p_i. For example, if we want to rank CEW-DTW value, r_{p_i} is $\boldsymbol{CEW-DTW}(X_i, Y)$. Therefore, according to Baltrunas et al. [64], the Discounted Cumulative Gain (DCG) is defined as following:

$$DCG_k = r_{p_1} + \sum_{i=2}^{k} \frac{r_{p_i}}{log_2(i)}. \qquad (2-6)$$

By applying (2-6), the normalized DCG (nDCG) is defined as following:

$$nDCG_k = \frac{DCG_k}{IDCG_k}, \qquad (2-7)$$

where, $IDCG_k$ is the maximum possible gain value, when we optimally re-order the k items in $p_1, p_2, \dots, p_k$.

Since we have obtained CEW-DTW rank, we will use it to calculate DCG. However, under the simplest ideal condition, we usually believe that the best inquired answer is the one that contains the greatest number of key phrases. Therefore, we rank zero/one vectors in terms of the number of 1 value in a vector. If one vector has more 1 value, it will be put forward in the rank. We also call the ranked sequence as the original order statistics sequence. The definition of order statistics is as follows:

$$Suppose\ x_1, x_2, \dots, x_k\ to\ be\ a\ sequence, the\ order\ statistics\ is:\ x_{(1)}, x_{(2)}, \dots, x_{(k)},$$

where, $x_{(1)} = \min\{x_1, x_2, \dots, x_k\}$ is called smallest order statistic; $x_{(k)} = \max\{x_1, x_2, \dots, x_k\}$ is called largest order statistic.

Ideal DCG (IDCG) does not mean DCG of ideal ranking. Jurgens and Klapaftis [65] show that Ideal DCG (IDCG) is obtained by sorting the weight of DCG items. Tiun et al. [66] mention that the keyword frequency indicates how frequent the particular concept is mentioned in the document. They determine that the higher the frequency, the more important the concept is deemed to be. So, we can let the word frequency represent the weight of sentences. The IDCG is usually obtained by manually operating DCG rank to an "ideal" situation. However, if the dataset is very huge, it is impossible to manually rank those sequences. It means we can consider a manual-operation sequence to be a hidden sequence, or we can call it an "ideal" ranking. Then, we can verify how the actual ranking is closer to this "ideal" ranking by applying (2-7). Among other choices, we try to use CEW-DTW to rank data. Here, we consider the keyword frequency to be the weight. According to the common sense, if an answer contains more keywords, this answer can express more useful information. Therefore, if the frequency of keywords is higher, the sentence weight is higher.

2.5.2 Actual Case Evaluation

We use Amazon data to evaluate our model. We select some keywords as examples to compare the ranking nDCG of CEW-DTW, DTW-D and DTW. We develop multiple-line charts to show

evaluation results. In these charts, X axis represents different tests for keywords; Y axis represents nDCG value. For example, if we choose following keywords:

{"metal","hole","holes","plastic","truck","installation","rear","model","item","car","vehicle","factory","box","cars","Ford","site","Amazon","kit","product","price","vehicles","website","store","system","problem","problems","tire","tires","bumper","weight","bolts","bottom","trailer","mirror","seat","key","paint","gas","oil","application","filter","wire","instructions","battery","power","OEM","batteries","light","Honda","lights","tailgate","roof","engine","motor","valve","cap","fuel","tank","sensor","Toyota","leather","chains"}. We provide some results in following, other results are similar:

Figure 1: Keyword: vehicle for DTW, DTW-D, and CEW-DTW

Test No.	CEW-DTW	DTW-D	DTW
1	0.995	0.913	0.913
2	1.0	1.0	1.0
3	0.999	0.924	0.923
4	0.942	0.908	0.908
5	0.973	0.819	0.819
6	1.0	1.0	1.0
7	0.998	0.965	0.965
8	1.0	1.0	1.0
9	0.887	0.888	0.888
10	0.9	0.813	0.813

Table 1: nDCG Value of DTW, DTW-D and CEW-DTW--- Keyword: vehicle

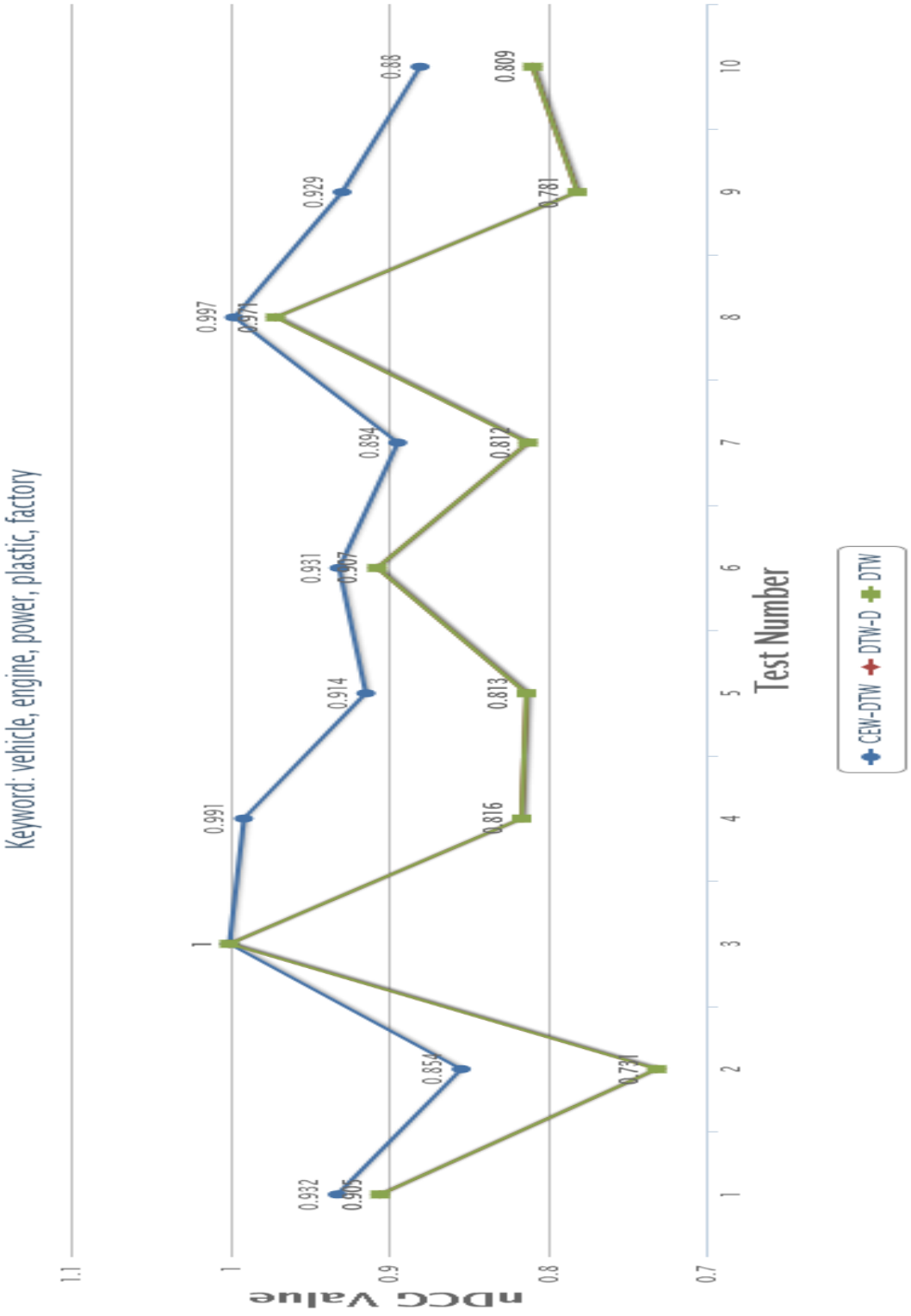


Figure 2: Keyword: vehicle, engine, power, plastic, factory for DTW, DTW-D, and CEW-DTW

Test No.	CEW-DTW	DTW-D	DTW
1	0.932	0.905	0.905
2	0.854	0.731	0.731
3	1.0	1.0	1.0
4	0.991	0.816	0.816
5	0.914	0.813	0.813
6	0.931	0.907	0.907
7	0.894	0.812	0.812
8	0.997	0.971	0.971
9	0.929	0.781	0.781
10	0.88	0.809	0.809

Table 2: nDCG Value of DTW, DTW-D and CEW-DTW --- Keyword: vehicle, engine, power, plastic, factory

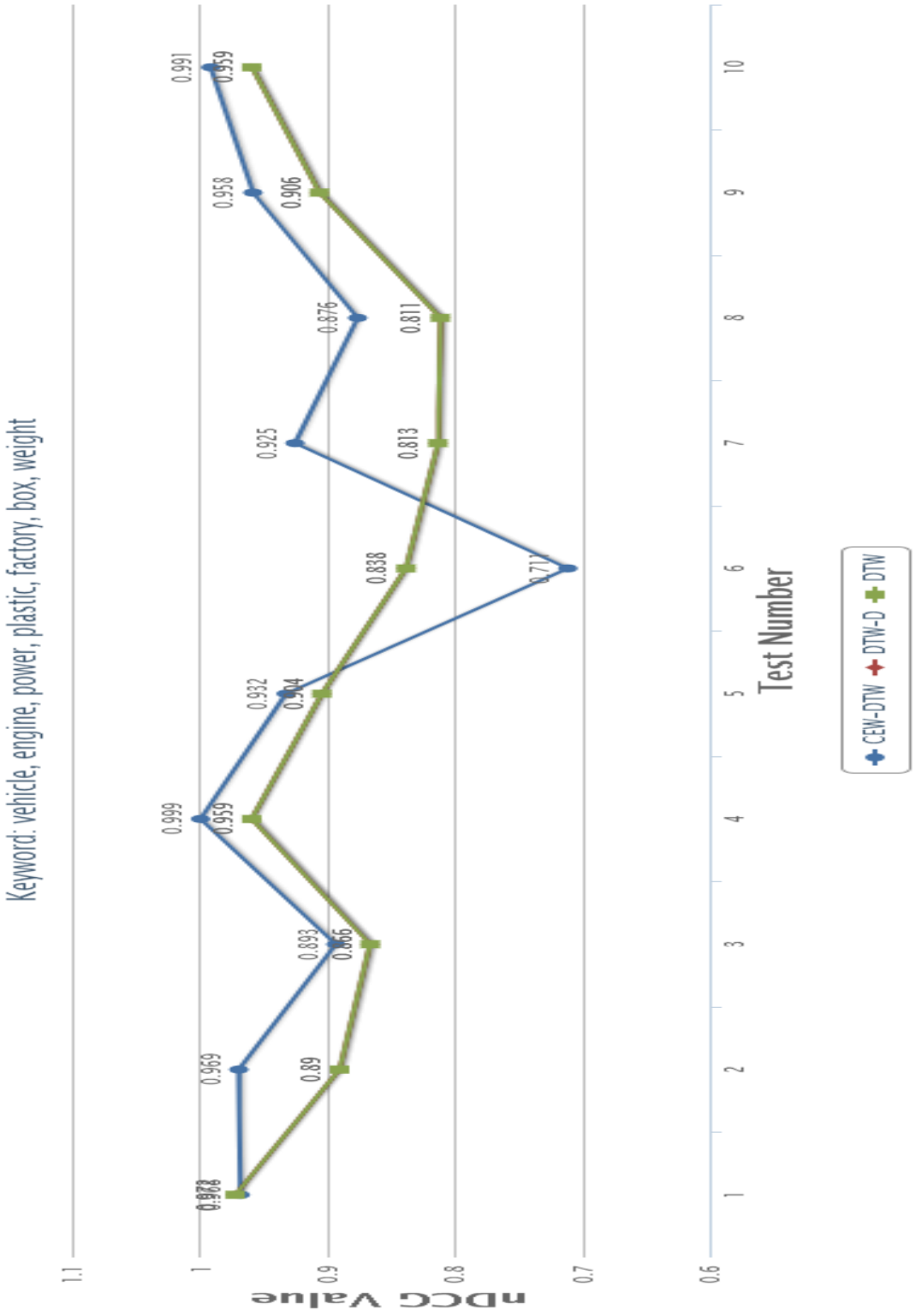


Figure 3: Keyword: vehicle, engine, power, plastic, factory, box, weight for DTW, DTW-D, and CEW-DTW

Test No.	CEW-DTW	DTW-D	DTW
1	0.968	0.972	0.972
2	0.969	0.89	0.89
3	0.893	0.866	0.866
4	0.999	0.959	0.959
5	0.932	0.904	0.904
6	0.711	0.838	0.838
7	0.925	0.813	0.813
8	0.876	0.811	0.811
9	0.958	0.906	0.906
10	0.991	0.959	0.959

Table 3: nDCG Value of DTW, DTW-D and CEW-DTW --- Keyword: vehicle, engine, power, plastic, factory, box, weight

2.5.3 Discussion

Moturu and Liu [67] demonstrate that a high nDCG value represents the high accuracy. Lee et al. [68] apply nDCG to test the performance of the thread ranking task. They demonstrate that the high nDCG score represents a high-quality rank than a low-quality one. Lee et al. [69] demonstrate that relevant documents show a higher nDCG evaluation score than nonrelevant documents. Therefore, in our research, the higher the nDCG value, the better the ranking quality.

In the above evaluation results figures (Figure. 1, Figure. 2, Figure. 3), we can clearly see that almost all nDCG scores of CEW-DTW are higher than those of other models. These figures demonstrate that the ranking quality of CEW-DTW is better than that of DTW-D and DTW separately. nDCG value of DTW-D and DTW are almost the same. It means that these two models cannot illustrate a clear difference in ranking. In Figure. 1, some points show that nDCG of CEW-DTW, DTW-D and DTW are the same because all answers do not contain keywords and all answer vectors are zero vectors. In Figure. 3, one nDCG value of CEW-DTW is worse than that of DTW-D and DTW separately, it illustrates that the CEW-DTW ranking quality is worse than other models'. Since the group nDCG mean value can reflect the overall situation of this group, we can describe the performance of these groups by calculating the mean value of nDCG.

Keywords	CEW-DTW	DTW-D	DTW
vehicle	0.969	0.922	0.922
vehicle, engine, power, plastic, factory	0.932	0.854	0.854
vehicle, engine, power, plastic, factory, box, weight	0.922	0.892	0.892

Table 4: Average nDCG of CEW-DTW, DTW-D, DTW in different keywords

Overall, the Table 4 clearly demonstrates that the average ranking nDCG value of CEW-DTW is better than that of DTW-D or DTW separately. Though one nDCG value of CEW-DTW in Table 4 is slightly worse than those of DTW-D or DTW in some keywords, it will not affect the average CEW-DTW ranking performance.

2.6 Conclusion of this chapter

We investigate answer ranking research in Amazon answer dataset. We explore the problem of time sequences relationship between actual answers and an "ideal" answer. We apply the Entropy method to remove noise as many as possible to highlight useful information. We propose a new

model CEW-DTW to rank answers. Additionally, the popular assessment rules: nDCG is applied to verify the ranking quality. Compared with DTW and DTW-D, the new model CEW-DTW shows an obvious improvement of the ranking quality.

Chapter 3

3 A new Kullback-Leibler based model to analyze texts with at least one keyword

Text ranking is a popular research field. Many researchers have developed methods to find answers of high qualities. Qualities are represented by ranks of the answers according to a certain evaluation standard. We have developed CEW-DTW model to rank answers based on their distances to the “ideal” answer. The distances are determined by the frequencies of keywords. However, it lacks the distance (divergence) information of distributions of noise and keywords to the "ideal" answer. In this chapter, we develop a new model called KL-CEW-DTW by incorporating Kullback-Leibler divergence into the distance. This model does not only consider the time series of noise and keywords but also involves the distributions of noise and keywords. We use the standard of nDCG to test our model. We conclude that KL-CEW-DTW has a better performance than other models.

3.1 Background Introduction of this Chapter

Kullback-Leibler divergence is considered as a statistics method, which plays an important role in information analysis (Raiber and Kurland [70]) for text data analysis and machine learning. Kullback-Leibler divergence mainly measures the difference of two distributions. It can assess answers in terms of a weighted geometric mean.

Since people have different viewpoints about text qualities, keywords as an important feature are usually employed in analysis of text qualities. Intuitively, a proper ranking method should be able to assign higher ranks to answers containing more keywords. However, using only the number of keywords is insufficient to reflect the difference between an answer and the “ideal” answer, which may lead to the failure of reflecting text quality accurately. Therefore, a better ranking method should also reflect the spread of keywords in answers, which could be characterized by keywords distributions, particularly those with a well-defined probability density function. The method combines the classical kinetic analysis and risk calculation method using probability density

function (Oya [71]). In text ranking, the Kullback-Leibler divergence has been regarded as a popular method to evaluate text quality from the viewpoint of probability density (Raiber and Kurland [70]). Therefore, we would like to use this methodology to improve CEW-DTW.

3.2 Literature Review

Kullback–Leibler divergence has been discussed in recent publications. Ponti et al. [72] present a decision cognizant Kullback–Leibler divergence model (DC-KL), which is proved to have a better discriminating statistical properties in pattern recognition systems. Bušic and Meyn [73] present a method to the problem pf MDPs. They choose a reward methodology to calculate the weight of parameter in MDPs. Raiber and Kurland [70] use Kullback-Leibler divergence to develop a language model for the assessment of the inverse document frequency. Some typical evaluations have proved that Kullback-Leibler divergence performs efficiently when parameters are alternative. Ha et al. [74] apply a Kullback-Leibler restraint to improve the estimate stability in the research of spectrum estimation of x-ray. Their algorithm is proved to be an optimized way for the analysis of x-ray CT images. Galas et al. [75] apply Kullback-Leibler divergence in analysis of series of interacting variables in terms of the Möbius inversion duality. They develop a distance model to illustrate a metric under some restricted conditions. Based on the Kullback-Leibler divergence, Delpha et al. [76] present a method to find faults. They use data with Gamma distribution to assess this method and find that this method has a high accuracy of fault detection. Kullback-Leibler divergence is also used in the field of weather. Li et al. [77] present a KL distance-based DRO model to find uncertainties in weather forecasting. Compared with the robust optimization model, this model has less conservatism. Kullback-Leibler can also be applied in the image research field. Maddux et al. [78] use Kullback-Leibler method to find the similarity in different image data set. This method can help researchers to judge which category the new image belongs to. This method can also forecast immunogenicity of image.

3.3 Model

3.3.1 Kullback-Leibler divergence

According to the description of Johnson and Sinanovic [79], if we have two probability vectors, **P(x)**, **Q(x)**, we can write Kullback-Leibler divergence definition in the following way:

$$\boldsymbol{KL}_{Distance}(\boldsymbol{P}||\boldsymbol{Q}) = \int P(x) \times \log[\frac{P(x)}{Q(x)}]dx.$$

Kullback-Leibler divergence is not a symmetric distance because **P(x)** and **Q(x)** are in numerator and denominator respectively. $KL_{Distance}(\boldsymbol{P}||\boldsymbol{Q})$ can be intuitively understood as the distance from **P(x)** to **Q(x)**. In this formula, it is clearly that **Q(x)** cannot be zero.

When **P(x)** and **Q(x)** are discrete, we can define Kullback-Leibler divergence in another way. Suppose $\boldsymbol{P^X}(x)$: $\{P_1^X(x), P_2^X(x), \dots, P_n^X(x)\}$ and $\boldsymbol{Q^Y}(y)$: $\{Q_1^Y(y), Q_2^Y(y), \dots, Q_n^Y(y)\}$ are two discrete probability distributions, the Kullback-Leibler distance formula is as follows:

$$\boldsymbol{KL}(\boldsymbol{P^X}, \boldsymbol{Q^Y}) = \sum_{i=1}^{n} P_i^X(x) \times \log[\frac{P_i^X(x)}{Q_i^Y(y)}],$$

where, $\boldsymbol{Q^Y}$ cannot be zero. When $\boldsymbol{P^X}$ is zero, since we have

$$\lim_{t \to 0^+} t \times \log(t) = 0,$$

we can get that

$$\lim_{x \to 0^+} P_i^X(x) \times \log[\frac{P_i^X(x)}{Q_i^Y(y)}] = 0.$$

Since we use R package (e.g. entropy) to calculate Kullback-Leibler divergence, we can use ln(.) to replace log(.), the function will be transferred to the following way:

$$\lim_{x \to 0^+} P_i^X(x) \times \ln[\frac{P_i^X(x)}{Q_i^Y(y)}] = 0.$$

For example, let's give two probability density distributions, **V** and **W**, which are illustrated in the following chart. The red bar in the chart represents the distribution of the probability density **V**. The blue bar represents the distribution of the probability density **W**.

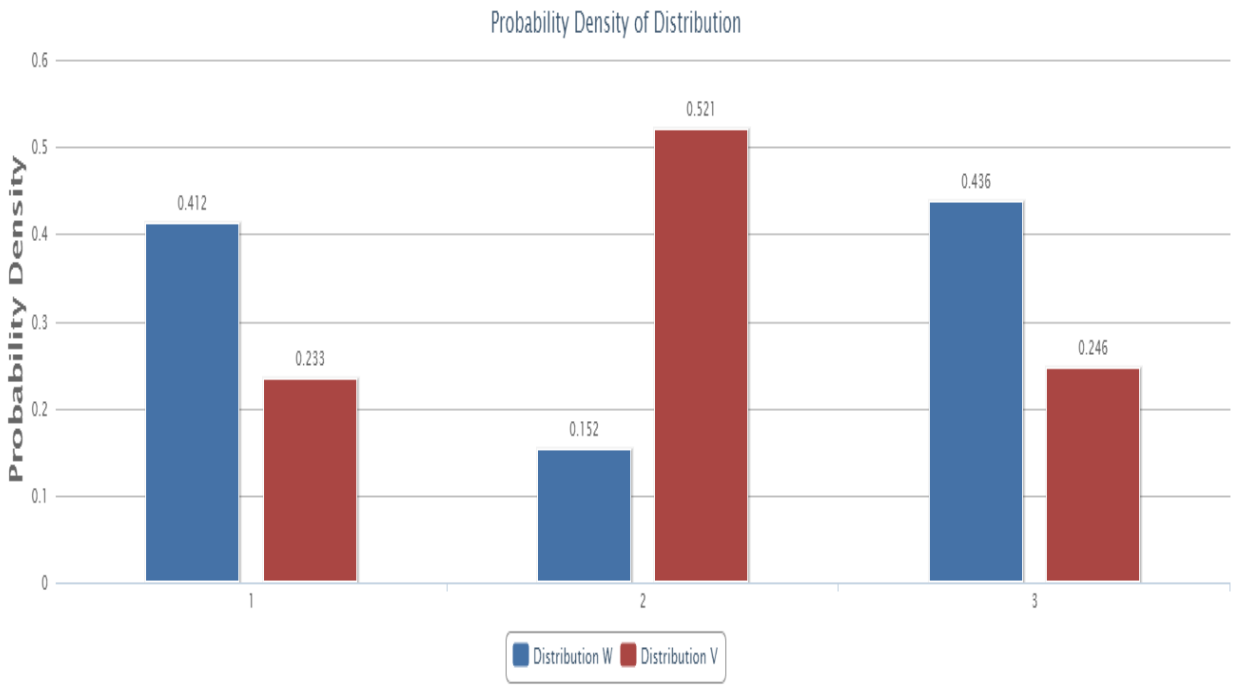


Figure 4: Density Distribution of Two Vectors.

No.	V	W
1	0.412	0.233
2	0.152	0.521
3	0.436	0.246

Table 5: Probability Densities of Two Vectors

Then, the Kullback-Leibler divergence (KL) of **V** and **W** is obtained as follows:

$$\boldsymbol{KL}(\boldsymbol{V},\boldsymbol{W}) = \sum_{i=1}^{3} \boldsymbol{V}(i) \times \ln\left[\frac{\boldsymbol{V}(i)}{\boldsymbol{W}(i)}\right]$$

$$= 0.412 \times \ln\left[\frac{0.412}{0.233}\right] + 0.152 \times \ln\left[\frac{0.152}{0.521}\right] + 0.436 \times \ln\left[\frac{0.436}{0.246}\right]$$

$$= 0.2348 - 0.1872 + 0.2495$$

$$= 0.2971$$

According to the description of KL.empirical function (Jean and Korbinian [80]), when we apply Kullback-Leibler divergence in zero/one vectors. We transfer these zero/one vectors to be probability density vectors firstly. Then, we will use KL.empirical function to calculate Kullback-Leibler divergence. For example, there are two vectors:

$$\mathbf{X}: \{1,1,1,1\}\ and\ \mathbf{Y}: \{0,1,0,1\}.$$

For the vector **X**, the probability density of element 1 is equal to 1, and the probability density of element 0 is 0. For the vector **Y**, the probability density of element 1 is 0.5 and the probability density of element 0 is 0.5. Thus, we have probability density of 0 and 1 as

$$\boldsymbol{P_X}: \{0,1\}\ and\ \boldsymbol{P_Y}: \{0.5,0.5\}.$$

Therefore, the Kullback-Leibler divergence of **X** and **Y** is 0.6931 (Here, KL.empirical uses ln(.) to replace log(.)). Similarly, if we have two vectors:

$$\mathbf{X}: \{1,1,1,1\}\ and\ \mathbf{Y}: \{0,0,0,1\},$$

the Kullback-Leibler divergence of **X** and **Y** is 1.3863.

For another example of two equal vectors, where:

$$\mathbf{X}: \{1,1,1,1\}\ and\ \mathbf{Y}: \{1,1,1,1\}.$$

For the vector **X**, the probability density of element 1 is 1 and the probability density of element 0 is 0. For the vector **Y**, the probability density of element 1 is 1 and the probability density of element 0 is 0. Thus, these two vectors have equal probability densities as follows:

$$\boldsymbol{P_X}: \{0,1\}\ and\ \boldsymbol{P_Y}: \{0,1\}.$$

Therefore, the Kullback-Leibler divergence of **X** and **Y** is 0. Since the theory of Kullback-Leibler suggests that large value of Kullback-Leibler divergence means that these two vectors are far away from each other, we would like to say that the above two vectors are the closest to each other.

3.3.2 KL-CEW-DTW

We combine Kullback-Leibler divergence and CEW-DTW together to be a new methodology: Kullback Leibler-Cosine Eudiean Warping-Dynamic Time Warping (KL-CEW-DTW). Since CEW-DTW analyze zero/one vectors, we also use these vectors in our new methodology. The formula of the new methodology is as follows:

$$\mathbf{KL-CEW-DTW}(\mathrm{X},\mathrm{Y})$$

$$= \mathbf{CEW-DTW}(\mathrm{X},\mathrm{Y}) + \mathbf{KL}(\mathrm{X},\mathrm{Y}), \qquad (3-1)$$

where, X is a zero/one vector, referred as an individual vector; Y is an "ideal" vector, with each element to be 1. $\mathbf{KL}(\mathrm{X},\mathrm{Y})$ is the Kullback-Leibler divergence of X and Y. Suppose we have n answer vectors, $X_i, i = 1,2,3,\dots,n$, then the formula will be rewritten to the following way:

$$\mathbf{KL-CEW-DTW}(X_i,\mathrm{Y})$$

$$= \mathbf{CEW-DTW}(X_i,\mathrm{Y}) + \mathbf{KL}(X_i,\mathrm{Y}). \qquad (3-2)$$

3.4 Assessment

In this section, we use the same data in Chapter 2 for the assessment and want to see whether KL-CEW-DTW is better than CEW-DTW. The rule of transferring answers to be zero/one vectors is also the same to that in Chapter 2. That is, we match each word of a selected answer with the group of keywords. If one word appears in the keywords group, it will be assigned to be value 1; otherwise, it is 0. We begin by calculating the value of KL-CEW-DTW between the zero/one vector of each answer and the vector of an "ideal" answer (see Chapter 2 about details of an "ideal" answer), then rank these values. Also, we use nDCG to compare the performance of KL-CEW-DTW and CEW-DTW. The results of comparison are as follows:

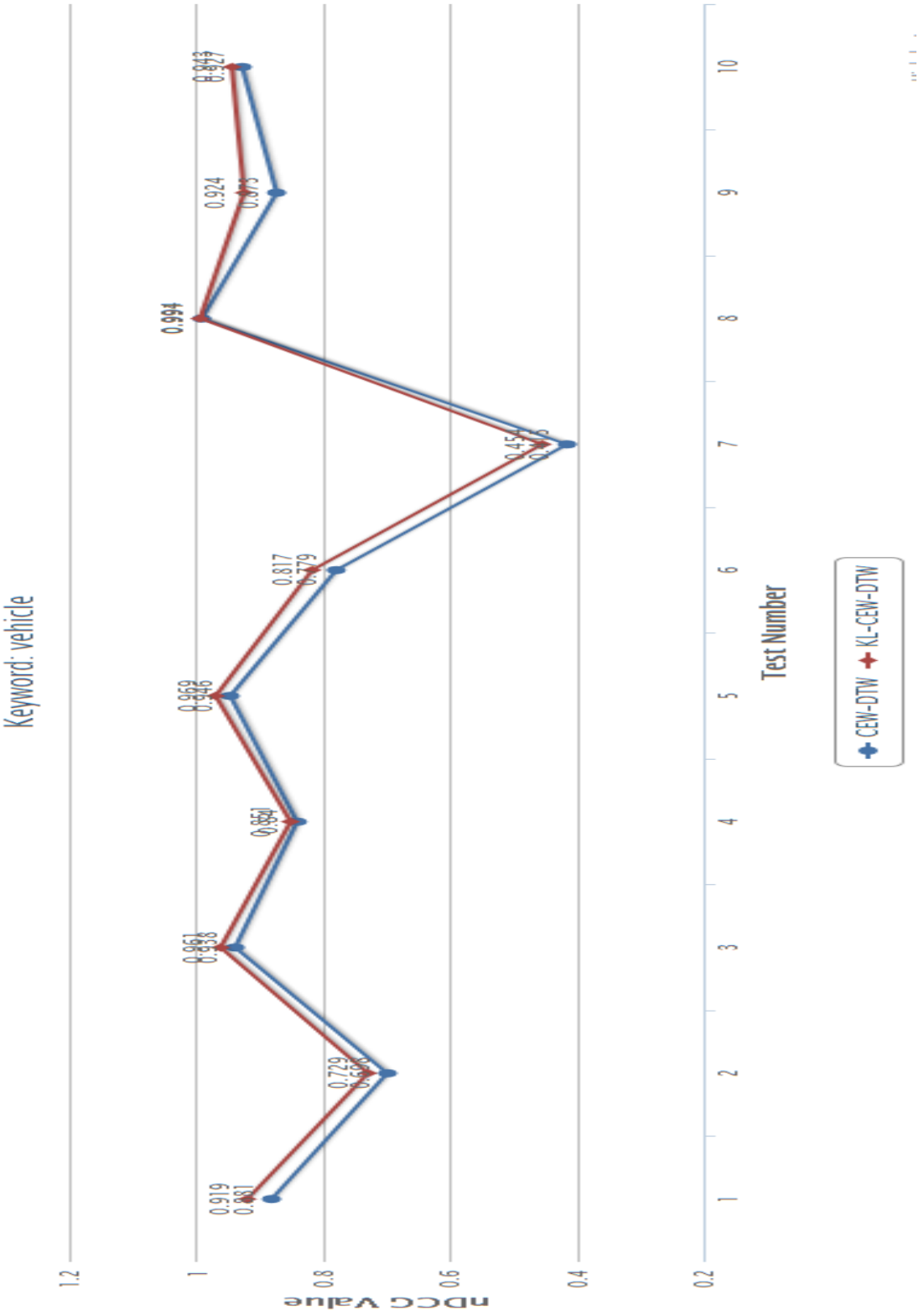


Figure 5: Keyword: vehicle for KL-CEW-DTW and CEW-DTW

Test No.	CEW-DTW	KL-CEW-DTW
1	0.881	0.919
2	0.698	0.729
3	0.938	0.961
4	0.840	0.851
5	0.946	0.969
6	0.779	0.817
7	0.415	0.454
8	0.991	0.994
9	0.873	0.924
10	0.927	0.943

Table 6: nDCG Value of CEW-DTW and KL-CEW-DTW--- Keyword: vehicle

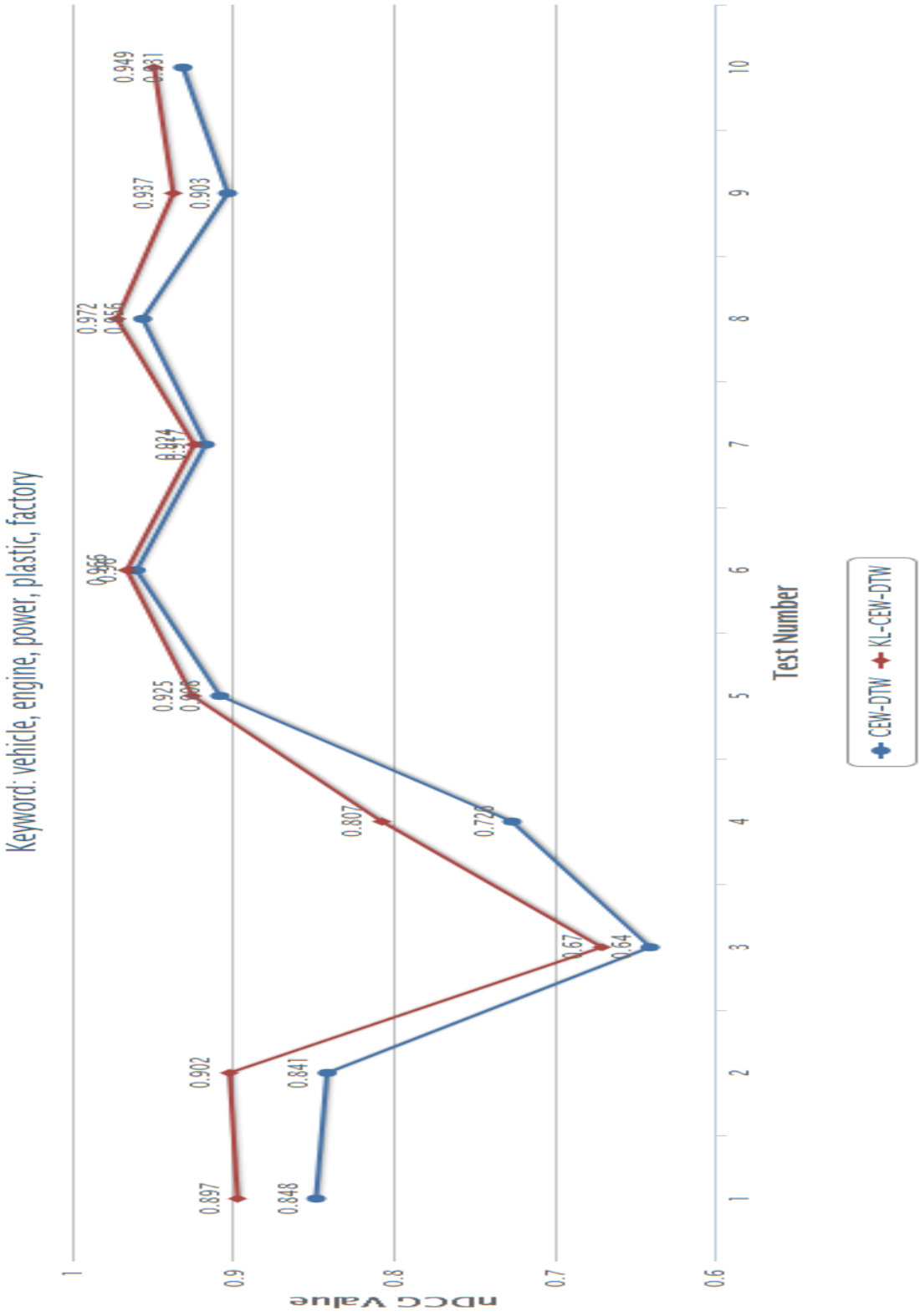


Figure 6: Keyword: vehicle, engine, power, plastic, factory for KL-CEW-DTW and CEW-DTW

Test No.	CEW-DTW	KL-CEW-DTW
1	0.848	0.897
2	0.841	0.902
3	0.640	0.670
4	0.726	0.807
5	0.908	0.925
6	0.960	0.966
7	0.917	0.924
8	0.956	0.972
9	0.903	0.937
10	0.931	0.949

Table 7: nDCG Value of CEW-DTW and KL-CEW-DTW--- Keyword: vehicle, engine, power, plastic, factory

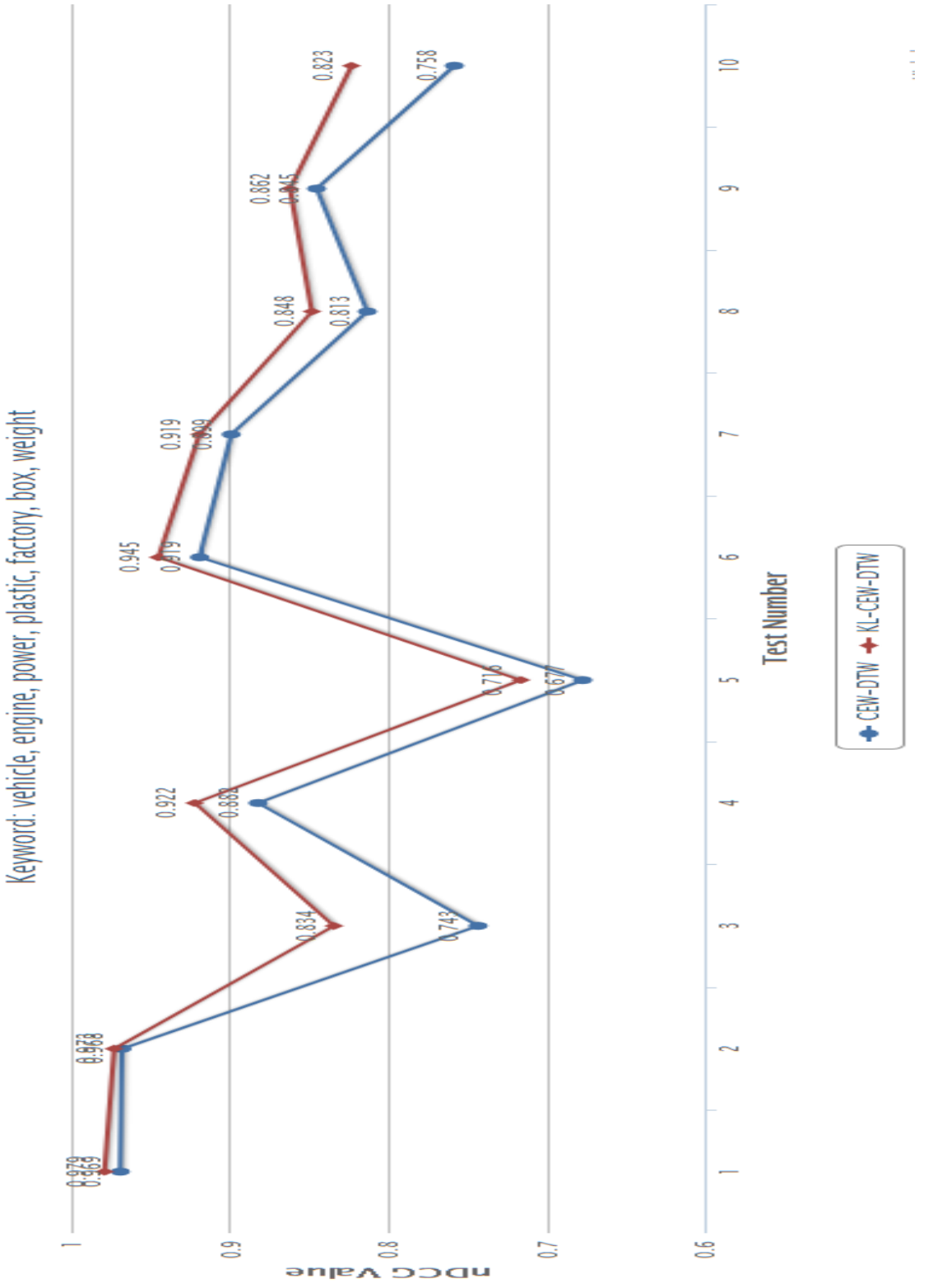


Figure 7: Keyword: vehicle, engine, power, plastic, factory, box, weight for KL-CEW-DTW and CEW-DTW

Test No.	CEW-DTW	KL-CEW-DTW
1	0.969	0.979
2	0.968	0.973
3	0.743	0.834
4	0.882	0.922
5	0.677	0.716
6	0.919	0.945
7	0.899	0.919
8	0.813	0.848
9	0.845	0.862
10	0.758	0.823

Table 8: nDCG Value of CEW-DTW and KL-CEW-DTW--- Keyword: vehicle, engine, power, plastic, factory, box, weight

3.4.1 Discussion

In the above Figures 5, 6, and 7, we can clearly see that almost all nDCG scores of KL-CEW-DTW are higher than CEW-DTW. These charts demonstrate that the ranking quality of KL-CEW-DTW is better than CEW-DTW. We can describe the performance of these groups by calculating the mean value of nDCG.

Keywords	CEW-DTW	KL-CEW-DTW

vehicle	0.829	0.856
vehicle, engine, power, plastic, factory	0.863	0.895
vehicle, engine, power, plastic, factory, box, weight	0.847	0.882

Table 9: Average nDCG of CEW-DTW and KL-CEW-DTW in different keywords

Overall, the Table 9 clearly demonstrates that the average ranking nDCG value of KL-CEW-DTW is better than that of CEW- DTW.

3.5 Conclusion

Though CEW-DTW is a good model for text ranking, it does not put the distribution of keywords into construction. In this chapter, we develop a new model KL-CEW-DTW. This model is based on CEW-DTW and still use the "ideal" answer as the assessment standard. KL-CEW-DTW can not only consider the time series of noise and keywords but also accounts for distributions of noise and keywords. KL-CEW-DTW is still assessed by the standard of nDCG. Assessment results show that KL-CEW-DTW performs better than CEW-DTW in ranking. KL-CEW-DTW can help people to rank answers from the viewpoint of keywords distribution with a better ranking quality than CEW-DTW. In practice, though KL-CEW-DTW cannot be applied in answers with all noise, it is a better choice to rank answers with at least one keyword.

Chapter 4

4 Probability Entropy

This chapter studies the answers retrieved from the Amazon questions discussed in earlier chapters, which have been analyzed based on keywords. Keywords and noise are defined by the frequency a word appears in a group of answers—those with high frequency are referred as keyword while the ones with low frequency are regard as noise. While keywords represent a set of single words, noise is a unique set that usually contains many words other than the keywords. For example, we may choose words with top-n frequency ranks to be the keywords. In this chapter, the elements of noise are considered as not distinguishable from each other, which will be discussed in later sections of this chapter.

Text noise includes unknown words, errors, and poor grammatical words composition. Current research directions, such as Information Retrieval/Extraction, Text classification/clustering, and Text mining provide methods for the analysis of both crucial information and noise. As far as we know, only a few of them consider using texts mining techniques to analyze both noise and keywords. Most researches usually use the number of keywords to represent the quality of a text. Generally, we consider that the more keywords an answer contains, the higher the quality this answer has. However, since noise can also reflect the text quality, our research will identify text quality using both keywords and noise.

4.1 Literature Review for this chapter

Information retrieval is a popular research topic with a large amount of literature. Mohan et al. [81] use deep learning to develop a model to retrieve information from biomedical literatures. Compared with NLP approaches, their model adapts word embedding approaches to select Delta features. Zhai and Lafferty [82] develop a language model to smooth documents by reconstructing the query-likelihood retrieval model. They combine some heuristics, such as TF-IDF and document length normalization, with a general retrieval formula to be a new model. Their model

shows a more sensitive performance in smoothing long documents than concise title documents. Turtle and Croft [83] present a framework using conditional probability for the incorporation of literature representations and strategies regarding the development of searching technology. Berger and Lafferty [84] propose a probabilistic approach from statistical machine translation to develop information retrieval methodologies. They present two methodologies to inquiry documents in translation processes, which both perform better than standard baseline vector space methodologies. Yoon et al. [85] adapt cosine similarity and pseudo-expansion to design a new method to retrieve information from news corpus. Experiment results illustrate that the new information retrieval method is helpful to create a corpus, which is close to news articles. Xu and Croft [86] analyze performance of three automatic query expansion methods about corpus. They illustrate that feedback and analysis from local documents perform more efficiently than those from global documents.

Text noises have also been widely studied. Agarwal et al. [87] analyze noises to classify documents. They implement experiments to analyze different kinds of noises to find noises effects when they want to classify document. Apostolova and Kreek [88] illustrate a machine learning model for text noises with metrics. They use noisy historical data to analyze different kinds of noises. Their model also suggests that if we can artificially design text classification rules for noises, the prediction quality of the model will be improved. Nguyen and Patrick [89] illustrate a text mining model to analyze clinical data. They analyze noise types and develop a machine-learning-base system to handle frequent noises. This model efficiently identifies noises and decreases mistakes that people make when they read clinical reports manually. Li et al. [90] present a topic model CSTM to analyze short texts. This model uses common topics to collect background text noises and to classify texts. Assessment results show that CSTM performs better than existing topic models in traditional text classification tasks. Xiang et al. [91] develop a multi-ary steganographic methodology in terms of additive noises. This model can improve security when we use secret information. Patel and Diwanji [92] adapt page segmentation methods to extract information and detect noises in web pages. They use text density algorithm to enhance accuracy in noise detection. Also, their model reduces the mistakes of positive/negative value in URL detection.

4.2 Model

4.2.1 Data Preparation

Since our research mainly focus on digital data, we need to digitalize answers before analysis. When we obtain a group of answers, we firstly clean these answers (see Chapter 6 about details of data). These answers are called the **original answers**. We regard collection of all original answers together as the **global answer set**. Each answer in the global answer set is referred as the **individual** answer. In this chapter, we again use answers from Amazon as our objective answers. For example, we use answers of "baby" category for our analysis (see Chapter 1 about data description).

4.2.2 Digitalization of the Answers

Assume we have n keywords tagged with $\{1, \dots, n\}$ and every word in the set of noise tagged with 0. We match these keywords to individual words in each original answer. If one word in an original answer matches tag $i, i = 0,1,2, \dots, n$, the location of this word will be tagged with this number. Thus, one answer can be transferred to be a numeric vector, each element of this vector is a number tag.

4.2.3 Definition of Global Probability and Individual Probability

After digitalization, we can calculate probabilities of noise and keywords. For example, for the n selected keywords, we define the probabilities of global answers as $\{Q_0, Q_1, \dots, Q_n\}$, where Q_0 represents the global probability of noise; $Q_i, i = 1,2, \dots, n$ represents the global probability of corresponding keywords. For the global answer set, we can calculate the noise frequency and keyword frequency across all of its answers. Let global word frequency of noise and each keyword be $\{Num_0^G, Num_1^G, \dots, Num_n^G\}$ where G is a label representing the global answer set. We can define $\{Q_0, Q_1, \dots, Q_n\}$ as follows:

$$Q_0 = \frac{Num_0^G}{Num_0^G + Num_1^G + \cdots + Num_n^G},$$

$$Q_1 = \frac{Num_1^G}{Num_0^G + Num_1^G + \cdots + Num_n^G},$$

……

$$Q_n = \frac{Num_n^G}{Num_0^G + Num_1^G + \cdots + Num_n^G},$$

where $\sum_{i=0}^{n} Q_i = 1$. In most cases, the noise probability is larger than any of the keyword probabilities, hence in this chapter, our objective collection of documents satisfies $Q_0 > Q_i, i = 1,2, \dots, n$. If the global probability of one word is 0, we do not regard this word as a keyword. In this chapter, before we decide which words are keywords, we would like to rank the word frequency of all words in the objective collection of documents. Then, we choose words with top word frequency to be keywords. In these keywords, frequency of each word should be larger than 0. Thus, if we choose n keywords, probabilities of noise and keywords are arranged to be: $Q_0 > Q_1 > \cdots > Q_n > 0$, where Q_0 is the largest one in most cases and $Q_0 < 1$. Here, the volume of answer data should be as large as possible. Thus, we can avoid the global probability of a keyword to be 1. Because if the global probability of a keyword is equal to 1, it means answers only contain one word, so there is no necessarily to analyze these answers. If probabilities of two keywords are equal, we can use any order between two. Additionally, if the transition probability from one keyword to another keyword is high, these two keywords may be regarded as only one keyword. Chapter 5 discuss the merging of two keywords.

Analogously, we define the probabilities of an individual answer as $\{P_0, P_1, \dots, P_n\}$, where P_0 represents the probability of noise in this answer; $P_i, i = 1,2, \dots, n$ represents the probability of keywords. For an individual answer, we can also calculate the noise frequency and keyword frequency. Let word frequency of noise and each keyword be $\{Num_0^I, Num_1^I, \dots, Num_n^I\}$, where I is a label representing this individual answer. We can define $\{P_0, P_1, \dots, P_n\}$ as follows:

$$P_0 = \frac{Num_0^I}{Num_0^I + Num_1^I + \cdots + Num_n^I},$$

$$P_1 = \frac{Num_1^I}{Num_0^I + Num_1^I + \cdots + Num_n^I},$$

……

$$P_n = \frac{Num_n^I}{Num_0^I + Num_1^I + \cdots + Num_n^I}.$$

where $\sum_{i=0}^{n} P_i = 1$. Notice that: some P_i could be 0 and $P_0 > P_1 > \cdots > P_n$ may not hold.

Here, we give an example about how to get P_i and Q_i as following:

Example Keywords:	fit, seat
Answer 1	gate plus extension fit well inch opening concerned max fit well
Answer 2	wide base seat trying find booster fit between car-seats

Step 1. Get words frequency of noise and keywords respectively

- Total words frequency of noise and keywords respectively

	noises	keyword: “fit”	keyword: “seat”
Frequency	16	3	1

- Words frequency of noise and keywords respectively in each answer

	noises	keyword: “fit”	keyword: “seat”

Frequency	Answer 1	9	2	0
	Answer 2	7	1	1

Step 2. Get P_i and Q_i respectively

Q_0	Q_1	Q_2
0.8	0.15	0.05

	P_0	P_1	P_2
Answer 1	0.818	0.182	0
Answer 2	0.778	0.111	0.111

4.3 Entropy

After digital data preparation, we can obtain probabilities of noise and keywords globally and for individual answers. Hence, we could calculate the entropy based on these probabilities to assess the qualities of these answers. In the field of statistics, entropy is a method for assessing the total qualified information. More information means larger entropy. The entropy was defined by Shannon [93]. Given k states, suppose that the probability of an event i is P_i, where $i = 1,2, \dots, k$. The entropy of this event in these states can be defined as:

$$E = -\sum_{i=1}^{k} P_i \times Log(P_i),$$

where, $\sum_{i=1}^{k} P_i = 1, 0 \leq P_i \leq 1$.

4.3.1 Probability Model

We use weights of each event to improve the above entropy formula. Since we analyze a group of answers in terms of keywords, we regard this group of answers as a whole group. We calculate the total keyword probabilities in this group. We refer these probabilities as global probabilities and regard them as weights in the new developed entropy methodology.

Entropy has been applied in existing literatures, but the research objectives of them are different from ours. Zhao and Liu [94] combine DA-VMFS and SP-Kmeans algorithms with the maximum entropy principle to analyze the clustering problem of texts. Btoush and Dawahdeh [95] apply Entropy principle to compress text files, they test several algorithms, such as LZW, Huffman, Fixed-length code (FLC), and Huffman after using Fixed-length code (HFLC), to see their performance. Abualigah et al. [96] adapt Entropy to test clustering diversity of texts. In their entropy methodology, they calculate the percentage of one document in a group of documents. Abbas et al. [97] compare Entropy and Kullback–Leibler divergence to test their performance, they apply the minimum-cross entropy method to calculate the maximum log-probability. In their research, they do not consider noises in the information. They also illustrate that the performance depends on available problems and information. He et al. [98] use linguistic operator and the entropy weight method to find attribute weights when making decision for linguistic multi-attribute groups. They use matrix of elements as parameters in entropy model. Revanasiddappa et al. [99] use Intuitionistic Fuzzy Entropy to select text features when they want to categorize texts. In their Entropy formula, they use match degree as the parameter. Zhang et al. [100] adapt the K-nearest neighbor algorithm to develop a weighted entropy method about extreme value. This model uses the percentage of sample data as the parameter of entropy. Zou [101] produce a maximum entropy model to do text classification. In this model, the training data is considered as a weight. Zhang et al. [102] develop an active learning method to classify texts with convolutional neural networks. They adapt Shannon Entropy to be a measure to test uncertainty. Romero et al. [103] adapt the derivational Entropy to study an Active Learning technology about how to choose informative samples in terms of HTR scenario. They use ranges over all possible label sequences

as input of entropy. Zheng et al. [104] apply Fuzzy C-Means and Information Entropy to develop a new PageRank Algorithm. They use PageRank weight as the parameter of entropy. Namazi et al. [105] use entropy to analyze the complex structure of Bulk Metallic Glasses. They apply entropy to find properties of BMG's compressive strength. Bierig and Chernov [106] approach a convergence theory to find the maximum value of entropy. They apply the Multilevel Monte Carlo method to estimate a sequence of moments to get the maximum value of entropy. Laleye et al. [107] develop a new algorithm to analyze speech signals. This algorithm combines rényi entropy with singularity exponents in each point of the signal. Kan and Gero [108] study the characterization of designing processes and analyze the potential of design spaces in terms of the information entropy value of empirical data. They use Shannon entropy to do this analysis and apply the probability of occurrence of each symbol in Shannon Entropy. In these Entropy literatures, they do not consider noise nor the application of it in the entropy model. Also, these literatures do not discuss the derivation of maximum entropy probabilities from global probabilities.

Suppose that we have selected n keywords for a global answer set. We calculate global probabilities of these keywords, $\{Q_1, Q_2, \dots, Q_n\}$ as well as the probability of the noise Q_0. Similarly, for each individual answer, we can calculate probabilities of n keywords in this answer, $\{P_1, P_2, \dots, P_n\}$ and the probability of the noise, P_0. Thus, for each individual answer, we can define the **General Entropy** as follows:

$$E_n(\boldsymbol{P}) = -\sum_{i=0}^{n} Q_i \times P_i \times Log(P_i)\,. \qquad (4-1)$$

This general entropy represents entropy rates from individual probabilities with respect to the global probability. High value of the general entropy indicates high information quality. Intuitively, an answer should contain keywords as well as noise. So, we need to consider keywords as well as noise when we want to assess answers. The general entropy contributes to assess answers from the global and individual probabilities of keywords and noise.

We can get the following propositions regarding the General Entropy:

Proposition 1: For any answers,

(1) if $0 < P_0 < 1$, then $E_n(\boldsymbol{P}) > 0$;

(2) if $P_0 = 1$, then $E_n(\boldsymbol{P}) = 0$;

(3) if $P_0 = 0$, $E_n(\boldsymbol{P}) \geq 0$.

(4) $E_n(\boldsymbol{P}) = 0$ if and only if there exists $0 \leq i \leq n$, such that $P_i = 1, and\ P_j = 0\ for\ j = 0,1, \dots, n, j \neq i$.

Proof:

(1) $E_n(\boldsymbol{P}) = -\sum_{i=0}^{n} Q_i \times P_i \times Log(P_i) = -Q_0 \times P_0 \times Log(P_0) - \sum_{i=1}^{n} Q_i \times P_i \times Log(P_i)$. Since $Q_0 > 0$ and $0 < P_0 < 1$, then $-Q_0 \times P_0 \times Log(P_0) > 0$. On the other hand, $-\sum_{i=1}^{n} Q_i \times P_i \times Log(P_i) \geq 0$, thus, $E_n(\boldsymbol{P}) = -Q_0 \times P_0 \times Log(P_0) - \sum_{i=1}^{n} Q_i \times P_i \times Log(P_i) > 0$.

(2) If $P_0 = 1$, since $\sum_{i=0}^{n} P_i = 1$, we have $P_1 = P_2 = \cdots = P_n = 0$, thus, $E_n(\boldsymbol{P}) = -\sum_{i=0}^{n} Q_i \times P_i \times Log(P_i) = 0$.

(3) When $P_0 = 0$, if the answer may contain different kinds of keywords but no noises, then $E_n(\boldsymbol{P}) \geq 0$; if the answer only has one kind of keyword but no noises, then $E_n(\boldsymbol{P}) = 0$.

(4) When $E_n(\boldsymbol{P}) = -\sum_{i=0}^{n} Q_i \times P_i \times Log(P_i)$=0, we have $-Q_i \times P_i \times Log(P_i) = 0\ for\ i = 1,2, \dots, n$. Since $Q_i > 0$, we get $P_i = 1\ for\ some\ 0 \leq i \leq n$ and $and\ P_j = 0\ for\ j = 0,1, \dots, n, j \neq i$. On the other hand, if $P_i = 1\ for\ some\ 0 \leq i \leq n$ and $and\ P_j = 0\ for\ j = 0,1, \dots, n, j \neq i$, we get $-Q_i \times P_i \times Log(P_i) = 0\ for\ i = 1,2, \dots, n$. Thus, $E_n(\boldsymbol{P}) = 0$.

Here, we give some remarks to better explain this proposition:

(1) If the noise probability of an answer is between 0 and 1, i.e. $0 < P_0 < 1$, it means this answer contains not only noise but also some keywords. For example,

(1-1) If this answer only contains one keyword. Therefore, the probability of this keyword is also between 0 and 1. The general entropy of this answer should be a positive value. The entropy value also illustrates the quality information about

keywords and noise. The explanation is similar if the number of keywords is larger than one in an answer.

(1-2) If an answer only contains noise without any keywords, the probability of noise is 1. So, the general entropy should be 0. It means that this answer contains the minimum entropy.

(2) if $E_n(\boldsymbol{P}) > 0$, P_0 may be equal to 0. For example, the collection of digitalized answers is {1,1,2}, {0,0,0}, where 0,1 and 2 represents noise, keyword 1 and keyword 2 respectively. Global probabilities are $Q_0 = 0.5$, $Q_1 = 0.333$, and $Q_2 = 0.167$. $Q_0 > Q_1 > Q_2$. $E_2(\boldsymbol{P})$ of the first answer is $0.066 > 0$, but $P_0 = 0$.

(3) If $E_n(\boldsymbol{P}) = 0$, P_0 may not be 1. For example, the collection of digitalized answers is {1,1,1}, {0,0,0,0}, where 0,1 represents noise, the keyword 1 respectively. Global probabilities are $Q_0 = 0.571$, $Q_1 = 0.429$. $Q_0 > Q_1$. $E_1(\boldsymbol{P})$ of the first answer is 0, but $P_0 = 0$.

Some answers only have one word. We give a proposition about such kinds of answers as follows:

Proposition 2: For any answers, if the length of each answer is 1, then $E_n(\boldsymbol{P}) = 0$.

Proof: If the length of an answer is 1, this answer only contains one word. This word is either the noise or a keyword. If this word is the noise, then $P_0 = 1$ and $P_1 = P_2 = \cdots = P_n = 0$, thus, $E_n(\boldsymbol{P}) = -\sum_{i=0}^{n} Q_i \times P_i \times Log(P_i) = 0$. If this word is $No.i$ keyword, then $P_0 = P_1 = \cdots = P_{i-1} = P_{i+1} = \cdots = P_n = 0$ and $P_i = 1$, thus, $E_n(\boldsymbol{P}) = -\sum_{i=0}^{n} Q_i \times P_i \times Log(P_i) = 0$. So, we get $E_n(\boldsymbol{P}) = 0$.

When the length of an answer is 1, the general entropy of this answer is always 0. It means no matter if the word is a noise or a keyword; the general entropy is always the minimum. We remove answers with one word from the global answer set. Therefore, our candidate answers always contain at least two words.

For the special situation that individual probabilities of noise and keywords of an answer are equal to its respective global probabilities, we refer the General Entropy as the **global entropy**. We define it as follows:

Definition 1: Suppose the number of keywords to be n and individual probabilities of noise and keywords of an answer to be $\{P_0, P_1, \dots, P_n\}$. If $Q_i = P_i \; for \; i = 0,1,2, \dots, n$, then

$$E_n(\boldsymbol{Q}) = -\sum_{i=0}^{n} Q_i \times P_i \times Log(P_i) = -\sum_{i=0}^{n} Q_i \times Q_i \times Log(Q_i), \qquad (4-2)$$

where $E_n(\boldsymbol{Q})$ is referred as the global entropy.

Global entropy does not actually show the high information quality, thus it can be used in fake answers preventions (see the discussion in the following content).

4.3.2 Maximum General Entropy

4.3.2.1 Why should we need the Maximum General Entropy?

In the previous chapters, CEW-DTW assess answers in terms of the distance to an "ideal" answer. Here, we also want to obtain an answer, which is similar to the "ideal" answer. We try to find an objective goal so as to judge actual answers with respect to this goal. When we get a collection of answers, we usually cannot find which answer is the best. From statistical viewpoint, we try to get a goal, which we can find the imaginary answer to be the maximum entropy. We use the general entropy $E_n(\boldsymbol{P})$ to obtain an answer, which may never match any one answer in the collection of answers. We can call this answer to be **the Maximum General Entropy** answer. We use maximum general entropy probabilities to explain this answer from the viewpoint of statistics.

4.3.2.2 The Maximum General Entropy Answers

The Maximum General Entropy Probabilities is defined as follows:

Definition 2: Given global probabilities $\{Q_0, Q_1, \dots, Q_n\}$, the maximum general entropy answers $\vec{B} \coloneqq [B_0, B_1, \dots, B_n]^T$ with $\sum_{i=0}^{n} B_i = 1$, are defined by

$$\vec{B} = \underset{\vec{P}}{\operatorname{argmax}} E_n(\boldsymbol{P}),$$

where, $\vec{P} \coloneqq [P_0, P_1, \dots, P_n]^T$ with $\sum_{i=0}^{n} P_i = 1$.

Then, we have the following theorem for $\vec{B}$.

Theorem 1: Suppose the number of keywords $n \geq 2$, then, there exist a unique maximum general entropy answers $\vec{B} \coloneqq [B_0, B_1, \dots, B_n]^T$ so that $\vec{B} = \underset{\vec{P}}{\operatorname{argmax}} E_n(\boldsymbol{P})$ and $B_i = e^{-1-\frac{\lambda_0}{Q_i}}, i = 0,1, \dots, n$, where $\lambda_0 > 0$ is a unique positive value and $\sum_{i=0}^{n} B_i = 1$.

Proof: When $\{Q_0, Q_1, Q_2, \dots, Q_n\}$ are given, in order to maximize $E_n(\boldsymbol{P})$**,** we can get a function as following:

$$f(P_0, P_1, \dots, P_n, \lambda) = -\sum_{i=0}^{n} Q_i \times P_i \times Log(P_i) + \lambda(1 - P_0 - P_1 -, \dots, -P_n).$$

If we want to make $\frac{\partial f}{\partial P_i} = -(Log(P_i) + 1)Q_i - \lambda = 0$, we can get $\hat{P}_i = e^{-1-\frac{\lambda}{Q_i}}$. We define $B_i \coloneqq \hat{P}_i$, then $B_i = e^{-1-\frac{\lambda}{Q_i}}$. Thus, we can use $B_i, i = 0,1, \dots, n$ to make $E_n(\boldsymbol{P})$ to be maximum.

Since $\sum_{i=0}^{n} \hat{P}_i = 1$, we get $\sum_{i=0}^{n} e^{-1-\frac{\lambda}{Q_i}} = 1$, we can get a function $g(\lambda) = \sum_{i=0}^{n} e^{-1-\frac{\lambda}{Q_i}} - 1$, then we get $g'(\lambda) = \sum_{i=0}^{n} -\frac{1}{Q_i} e^{-1-\frac{\lambda}{Q_i}} < 0, for\ \lambda \geq 0$. On the other hand, it also means that $g(\lambda)$ is monotone decreasing for $\lambda \geq 0$. Since $g(\infty) = -1$ and $g(0) = \frac{n+1}{e} - 1 > 0, for\ n \geq 2$. Thus, we can find a unique positive λ_0 to make $g(\lambda_0) = 0$. It means we can get unique $B_i = e^{-1-\frac{\lambda_0}{Q_i}}, i = 0,1, \dots, n$.

From **Theorem 1**, when we obtain the maximum general entropy answers, we can imitate the definition of "ideal" answer in Chapter 2 to define the maximum general entropy "ideal" answer as follows:

Definition 3: An answer with the maximum general entropy is defined as the Maximum-Entropy-"Ideal" answer.

This definition enables us to use the uniform standard to represent an answer with the maximum general entropy in the following account.

Different values of Q_i may correspond to different values of B_i. Some B_i are larger than Q_i. Others are not. For simplicity, we define the situation of $Q_i > B_i$ or $Q_i < B_i, 0 \leq i \leq n$, as follows:

Definition 4: Given Q_i and $B_i, 1 \leq i \leq n$. If $B_i < Q_i$, then the keyword i is called "demotion". If $Q_i < B_i$, then the keyword i is called "promotion". Also, if $Q_0 < B_0$, the noise is called "promotion". If $Q_0 > B_0$, the noise is called "demotion".

The reason to call a keyword promotion is from the viewpoint of maximum general entropy: the frequency of the keyword should be higher than the global frequency of this keyword. On the other hand, when $B_i < Q_i$, the importance of keyword i is less than the importance of global level with respect to the maximum general entropy. Hence, this keyword can be dropped. This gives us a way to select a proper number of keywords for a study. The details are shown in next definition.

The definition of the maximum general entropy probabilities shows the process about how to get these probabilities. **Definition 3** illustrates that the answer with such kinds of probability distribution is called as the Maximum-Entropy-"Ideal" answer. In reality, the Maximum-Entropy-"Ideal" answer may not really exist. However, it gives us a target that we can be guided to find answers, which are used to compare with the Maximum-Entropy-"Ideal" answer. Here, if the length of an answer is 1, it may not be available to use this theorem. Though this theorem requires

the number of keywords is larger than or equal to two, it is not too much restricted to our analysis. In many answers, the number of keywords is much more than two.

Since noises in an answer are usually more than keywords, we try to analyze the global probability of the noise, Q_0, and the maximum general entropy probability of the noise, B_0, in the following proposition.

Proposition 3: For the Maximum General Entropy answers $\{B_0, B_1, B_2, \dots, B_n\}$, if $Q_0 > e^{-1}$ and $\lambda_0 > 0$, then $Q_0 > B_0$.

Proof: Suppose we have a function as following:

$$f(\lambda_0) = e^{-1-\frac{\lambda_0}{Q_0}} - Q_0.$$

Because $f'(\lambda_0) = -\frac{1}{Q_0} e^{-1-\frac{\lambda_0}{Q_0}} < 0, for \lambda_0 \geq 0$, we can say $f(\lambda_0)$ is monotone decreasing. Then, since $Q_0 > e^{-1}$, we can get $f(0) = e^{-1} - Q_0 < 0$. Thus, we get $f(\lambda_0) < 0$. Since $f(\lambda_0) = e^{-1-\frac{\lambda_0}{Q_0}} - Q_0 = B_0 - Q_0$, it means $Q_0 > B_0$.

In the general collection of answers, the global probability of noise, Q_0, is usually larger than 0.368 (e^{-1}). This proposition tells us that if an answer is or is close to the Maximum-Entropy-"Ideal" answer, its noise should be demoted with respect to the component of the global probability of noise. It also matches the objective intuition that a better-quality answer should be a less-noise one. When we try to judge which answer has a high quality, we should choose answers with less noises in terms of this proposition.

Now, since we know $B_i = e^{-1-\frac{\lambda_0}{Q_i}}, for\ i = 0,1, \dots, n\ and\ \lambda_0 > 0$, we try to compare B_i and Q_i of keywords in terms of λ_0. We firstly define a function as following:

$$g(Q) = (-logQ - 1)Q, \qquad (4-2)$$

where, $0 < Q < 1$. When we choose a positive λ_0, where $0 < \lambda_0 < \max_{0<Q<1} g(Q)$, we can plot $g(Q)$ as follows:

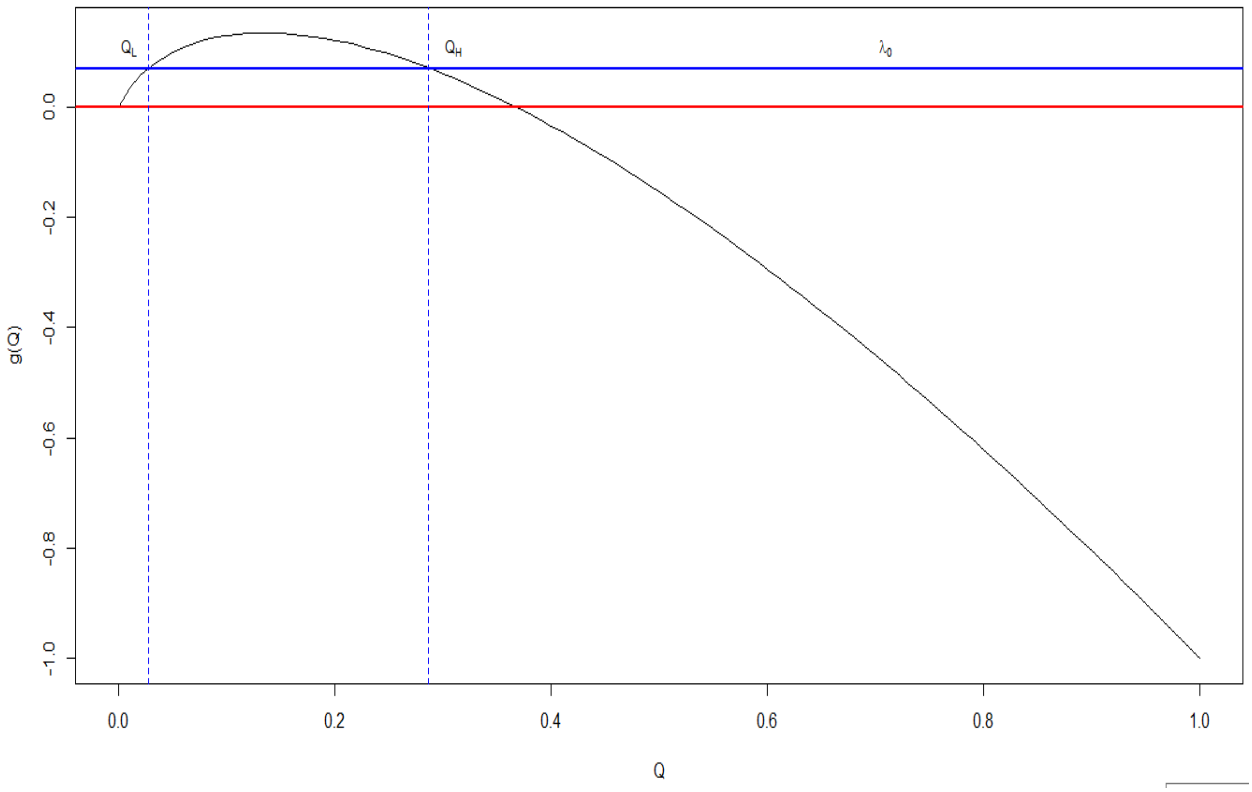


Figure 8: Function plot and cut-off value

The red line represents $g(Q) = 0$. The blue line represents λ_0. There must exist two points of intersection: the first one point is Q_L; the second one point is Q_H. Here, we give some remarks about B_i, Q_i, and λ_0:

(1) If $Q_L < Q_i < Q_H$, then we get $\lambda_0 < g(Q_i) = (-logQ_i - 1)Q_i$. We can deduce this inequation in following way:

$$\lambda_0 < (-logQ_i - 1)Q_i \Rightarrow \frac{\lambda_0}{Q_i} < (-logQ_i - 1)$$

$$\Rightarrow logQ_i < -1 - \frac{\lambda_0}{Q_i} \Rightarrow Q_i < e^{-1-\frac{\lambda_0}{Q_i}} = B_i.$$

(2) If $0 < Q_i < Q_L$ or $Q_H < Q_i$, then, we get $(-logQ_i - 1)Q_i = g(Q_i) < \lambda_0$. We can deduce this inequation in following way:

$$(-logQ_i - 1)Q_i < \lambda_0 \Rightarrow (-logQ_i - 1) < \frac{\lambda_0}{Q_i}$$

$$\Rightarrow -1 - \frac{\lambda_0}{Q_i} < logQ_i \Rightarrow B_i = e^{-1-\frac{\lambda_0}{Q_i}} < Q_i.$$

Here, we hope the probability of a keyword is as high as possible. However, if $Q_H < Q_i$, the keyword is demoted. We consider such a situation to be unreasonable. Because, if a keyword is repeated too many times, it may mislead readers to focus on this keyword and ignore other keywords. So, the probability of this keyword should not be too large. Generally, the situation of $Q_H < Q_i$ usually happens if the sample size of data is too small. So, our methodology is suggested to be applied in data with large sample size.

(3) Here, we give a table to show approximate value of Q_L and Q_H respectively with different value of λ_0

λ_0	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.1
Q_L	0.002	0.005	0.008	0.012	0.016	0.021	0.027	0.033	0.041	0.05
Q_H	0.358	0.347	0.337	0.325	0.314	0.301	0.289	0.275	0.26	0.244

From above remarks, when the global probability of a keyword is between Q_L and Q_H, the maximum general entropy answer of this keyword will be promoted from the global probability. Otherwise, it will be demoted from the global probability. In practice, though global probabilities of keywords are usually small, we can still find some keywords with global probabilities larger than Q_L. Therefore, we can give a definition about the optimum number of keywords as follows:

Definition 5: Among selected keywords from 1 to n, and a $0 < \lambda_0 < \max_{1 \le i \le n} g(Q_i)$ obtained in **Theorem 1**, the optimum number of keywords is defined as k, such that, $Q_L < Q_1, Q_2, \dots, Q_k < Q_H$ and $Q_{k+1}, Q_{k+2}, \dots, Q_n < Q_L$

This definition also illustrates that, those keywords with global probabilities smaller than Q_L should be converted to noises. Therefore, k can be considered as a cut-off number to determine the optimum number of keywords. Though we initially use top n keywords to digitalize answers, these answers will be re-digitalized if we finally have determined that the top k keywords are included. When we use top n keywords to digitalize answers, $Q_0, Q_1, Q_2, \dots, Q_n$ represent global probabilities of the noise, the keyword 1, the keyword 2, …, and the keyword n respectively. If we convert the keyword $k+1$, the keyword $k+2$, …, and the keyword n to be noises, global probabilities of the noise and keywords are $\widehat{Q_0}, \widehat{Q_1}, \widehat{Q_2}, \dots, \widehat{Q_k}$ respectively with $\sum_{i=0}^{k} \widehat{Q_i} = 1$. Also, $\widehat{Q_i} = Q_i$ for $i = 1,2, \dots, k$, $\widehat{Q_0} = Q_0 + \sum_{i=k+1}^{n} Q_i$. Correspondingly, we can use $\widehat{Q_i}, i = 0,1,2, \dots, k$ to calculate new $\widehat{B_i}$. Here, the value of λ_0 will change slightly. For simple notations, we can still use $Q_0, Q_1, Q_2, \dots, Q_k$ to represent $\widehat{Q_0}, \widehat{Q_1}, \widehat{Q_2}, \dots, \widehat{Q_k}$ respectively. Similarly, $B_0, B_1, B_2, \dots, B_k$ can be used to represent $\widehat{B_0}, \widehat{B_1}, \widehat{B_2}, \dots, \widehat{B_k}$. Thus, we can regard k to be the optimum number of keywords. Here, we can illustrate another remark about $E_n(\boldsymbol{P})$ as follows:

Remark: Since $Q_i, for\ i = 1,2, \dots, k$, does not change and λ_0 changes little, $B_i, for\ i = 1,2, \dots, k$, also changes little. When the value of Q_i is small, the value of B_i is also small. Therefore, if some keywords with very small global probabilities are removed, the overall change of $E_n(\boldsymbol{P})$ is slightly.

The question of the optimal number of keywords has been considered as a topic in many literatures. Dredze et al. [109] select nine keywords when they did keywords summary. They provide a short summary if the number of keywords in a document is less than nine. Wartena et al. [110] also illustrate the importance of the number of keywords. They think that a very small group of keywords do not result in the best recommendation. Thus, the number of keywords will affect the analysis quality of documents. Kommers et al. [111] illustrate that a limited number of keywords can speed up the searching performance. From these literatures, we know that too many or few

keywords are not good for us to analyze documents. So, we try to obtain the optimum number of keywords. This maximum general entropy answer gives us a way to find a cut-off number of optimum keywords. Maximum general entropy answers of top k keywords are promoted from their global probabilities respectively. We keep those keywords, maximum general entropy answers of which are promoted. Also, we throw away keywords, maximum general entropy answers of which are demoted. These keywords do not contribute too much to the information quality. Therefore, we can throw away these keywords from the perspective of maximum general entropy. We convert the keyword $k+1$, the keyword $k+2$, …, the keyword n to noises and still keep initial top k keywords. For original maximum general entropy answer $\{B_i\}, i = 0,1, \dots, n$, when $\{B_{k+1}, B_{k+2}, \dots, B_n\}$ are discarded, how much information are thrown away? In order to calculate the ratio of information entropy, we procedure a definition as following:

Definition 6: If the number of keywords is reduced from n to k, the corresponding relative efficiency of maximum general entropy answers, $\boldsymbol{P_{re}}$, can be defined to be:

$$\boldsymbol{P_{re}} = \frac{\sum_0^k \left(-log\widehat{B_{\iota,k}}\right)\widehat{B_{\iota,k}}}{\sum_0^n \left(-log B_{i,n}\right)B_{i,n}}, \qquad (4-3)$$

where, $\widehat{B_{0,k}}$ is the maximum general entropy answer of noise when the number of keywords is k; $\widehat{B_{\iota,k}}, 0 \le i \le k$ is the maximum general entropy answer of the keyword i when the number of keywords is k; $B_{0,n}$ is the maximum general entropy answer of noise when the number of keywords is n; $B_{i,n}, 0 \le i \le k$ is the maximum general entropy answer of the keyword i when the number of keywords is n.

4.3.2.3 Relationship between the Maximum General Entropy Answer of Noise (B_0) and the Global Probability of Noise (Q_0)

We can choose different λ_0 as examples (e.g. $\lambda_0 = 0.00867277$, $\lambda_0 = 0.01867277$, or $\lambda_0 = 0.02867277$) to analyze the relationship between B_0 and Q_0. If we choose $0.4 \le Q_0 \le 1.0$ to be examples, the relationship between Q_0 and B_0 is as following:

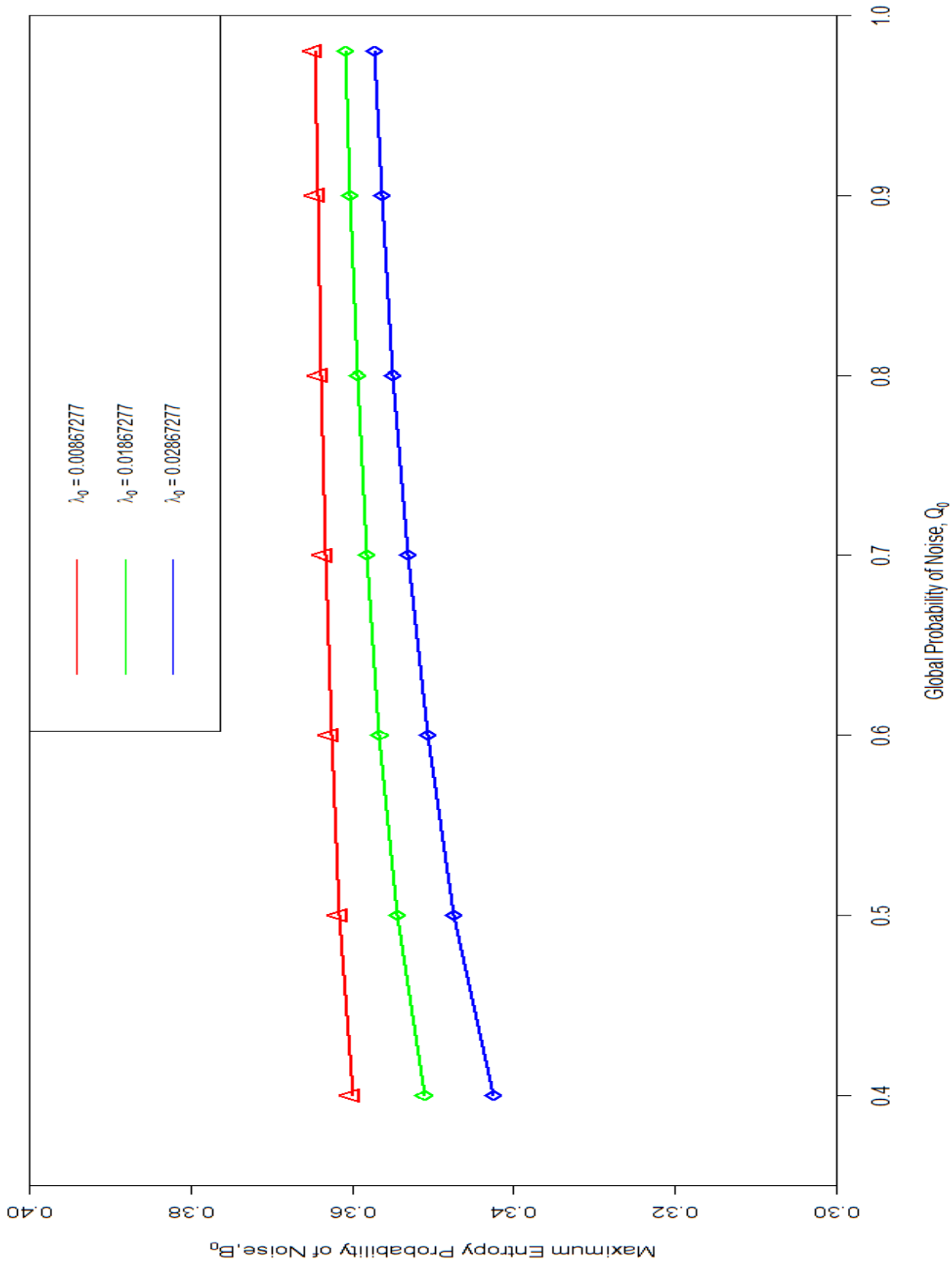


Figure 9: Relationship of Noise Probability

In Figure 9, the value of Q_0 ranges between 0.4 and 1.0, the value of B_0 is between 0.3 and 0.4. Here, $Q_0 > e^{-1} = 0.368$, thus, it is obviously that $Q_0 > B_0$. Though the value of λ_0 changes from 0.00867277 to 0.02867277, the range of variation of B_0 is almost similar when Q_0 is chosen between 0.4 and 1.0.

4.3.2.4 Relationship between the Maximum General Entropy Answer of Keywords (B_i) and the Global Probability of Keywords (Q_i)

Though we use many keywords for analysis, for simplicity, we choose two global keywords probabilities as examples. We want to show how different λ_0 affect the promotion from Q_0 to B_0. Since $0.4 \leq Q_0 < 1.0$, let $0 \leq Q_i \leq 0.2, i = 1,2$ (e.g. $Q_1 = 0.005, Q_2 = 0.015$). When we choose different λ_0 (e.g. $\lambda_0 = 0.00867277$, $\lambda_0 = 0.01867277$, or $\lambda_0 = 0.02867277$), we can find relationship between B_i and Q_i. The distribution between Q_i and B_i is as follows:

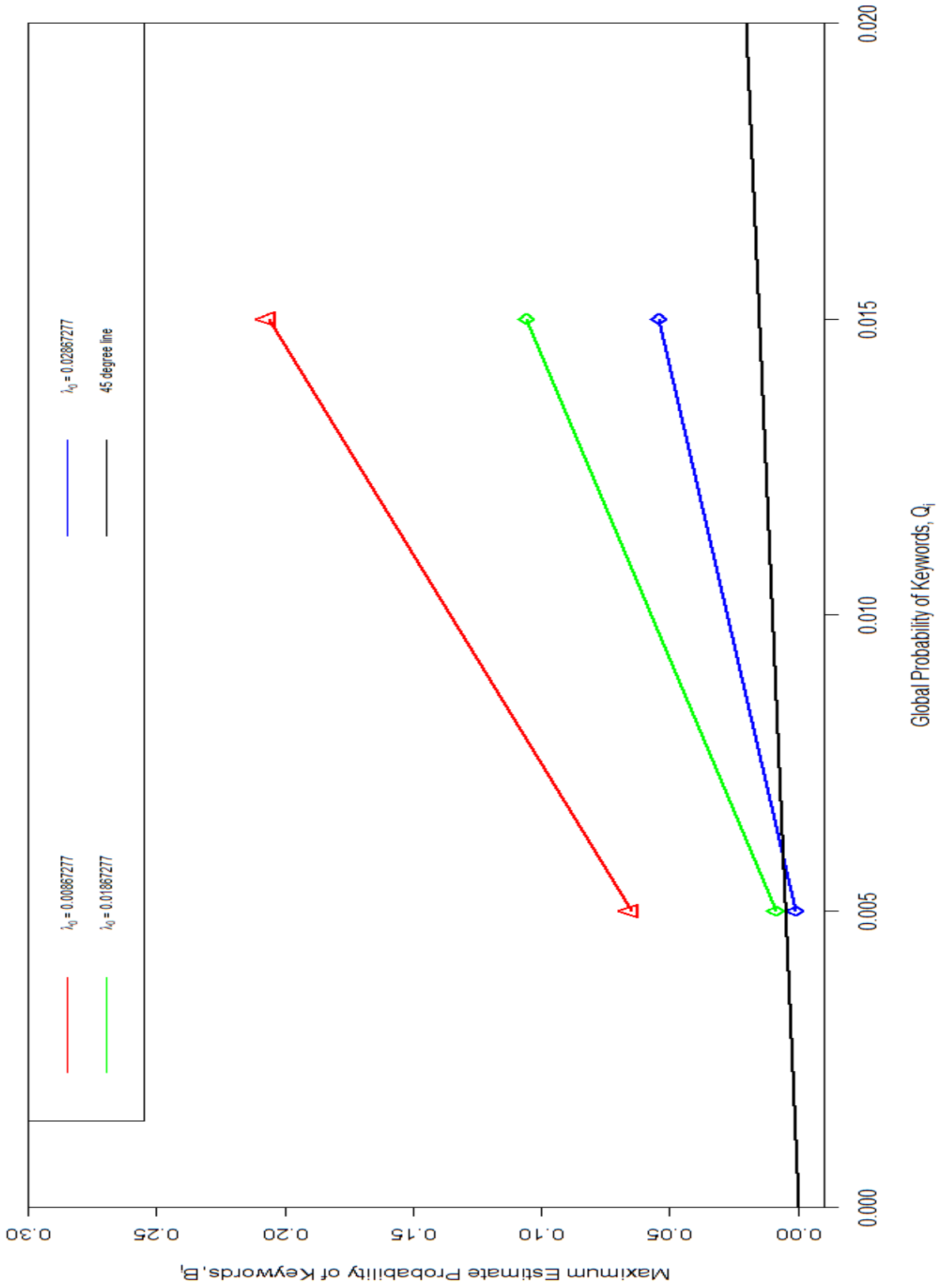


Figure 10: Relationship of Keywords Probability

In Figure 10, we find that different λ_0 can bring different promotion to B_i from Q_i in terms of different Q_i. Larger Q_i gives more promotion of B_i. If Q_i is very small, when λ_0 is high, B_i will be demoted from Q_i.

4.3.3 Application in Amazon data

In this chapter, we choose answers of "baby" category in Amazon data as an example. After digital data preparation, we initially choose as many keywords as possible in terms of their word frequencies. However, since the number of these keywords may be too many to analyze actual answers, we try to determine a suitable number of keywords for our analysis. We initially choose n words as keywords with global probabilities $\{Q_1, \dots, Q_n\}$, and we can calculate maximum general entropy answers $\{B_1, \dots, B_n\}$ in term of these global probabilities. We can decide the number of keywords by comparing Q_i and $B_i, i = 1,2, \dots, n$. For example, we firstly select 100 words to be keywords with global probabilities $\{Q_1, \dots, Q_{100}\}$ and maximum general entropy answers $\{B_1, \dots, B_{100}\}$. Their relationship is as follows:

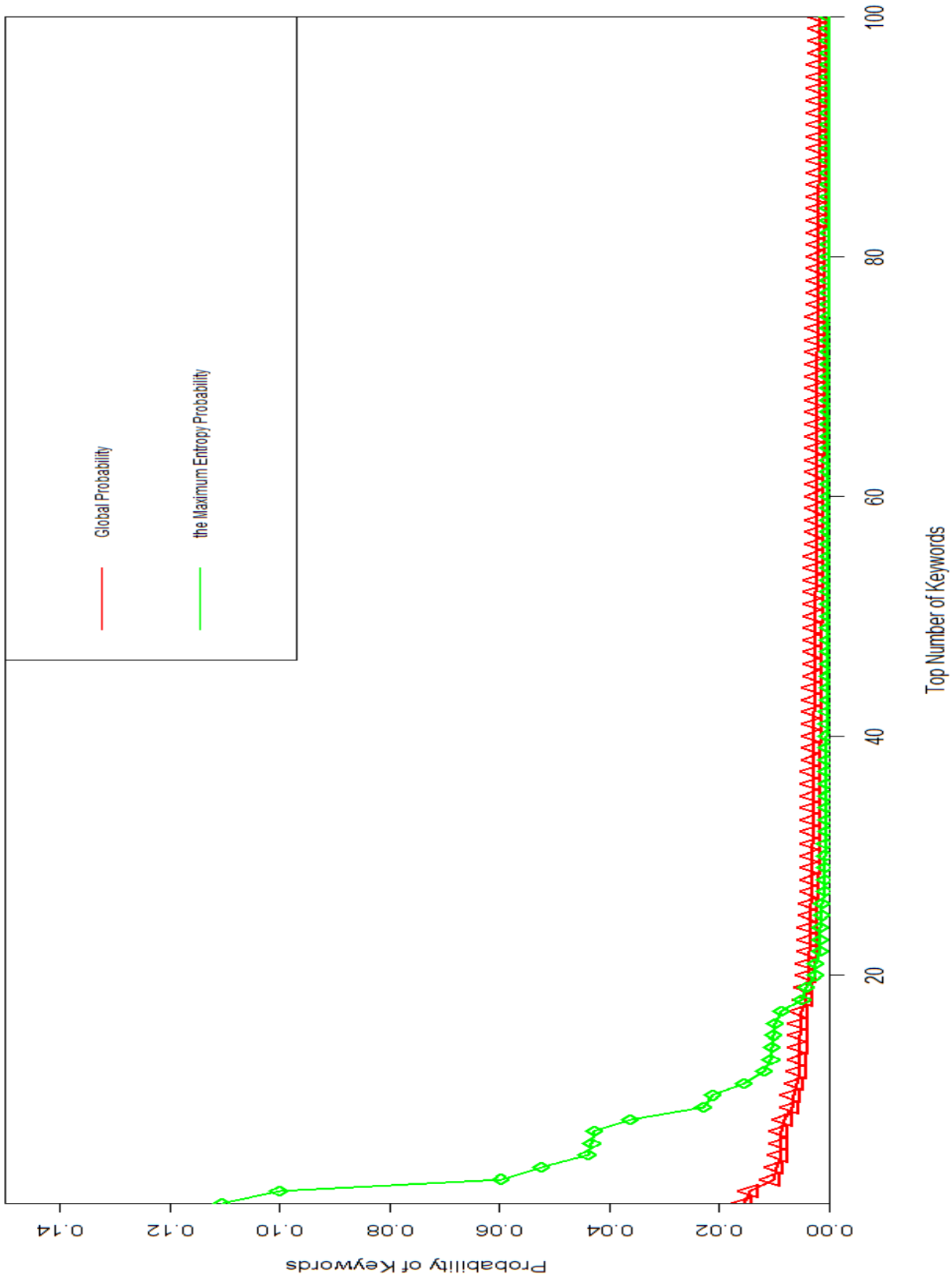


Figure 11: Relationship of Top 100 Keywords Probability

According to the formula of the maximum general entropy answer, when the global probability of a keyword becomes small enough, there is no obvious promotion for the maximum general entropy answer of this keyword. Thus, we can regard this keyword as noise. In this figure, the red line represents global probabilities of keywords and the green line represents maximum general entropy answers of keywords. We can use **Definition 5** to find that the optimum number of keywords is 19. Therefore, if the tag of the keyword is larger than 19, there is no promotion of this keyword. If we choose these top 19 keywords, the relationship between global probabilities and maximum general entropy answers is:

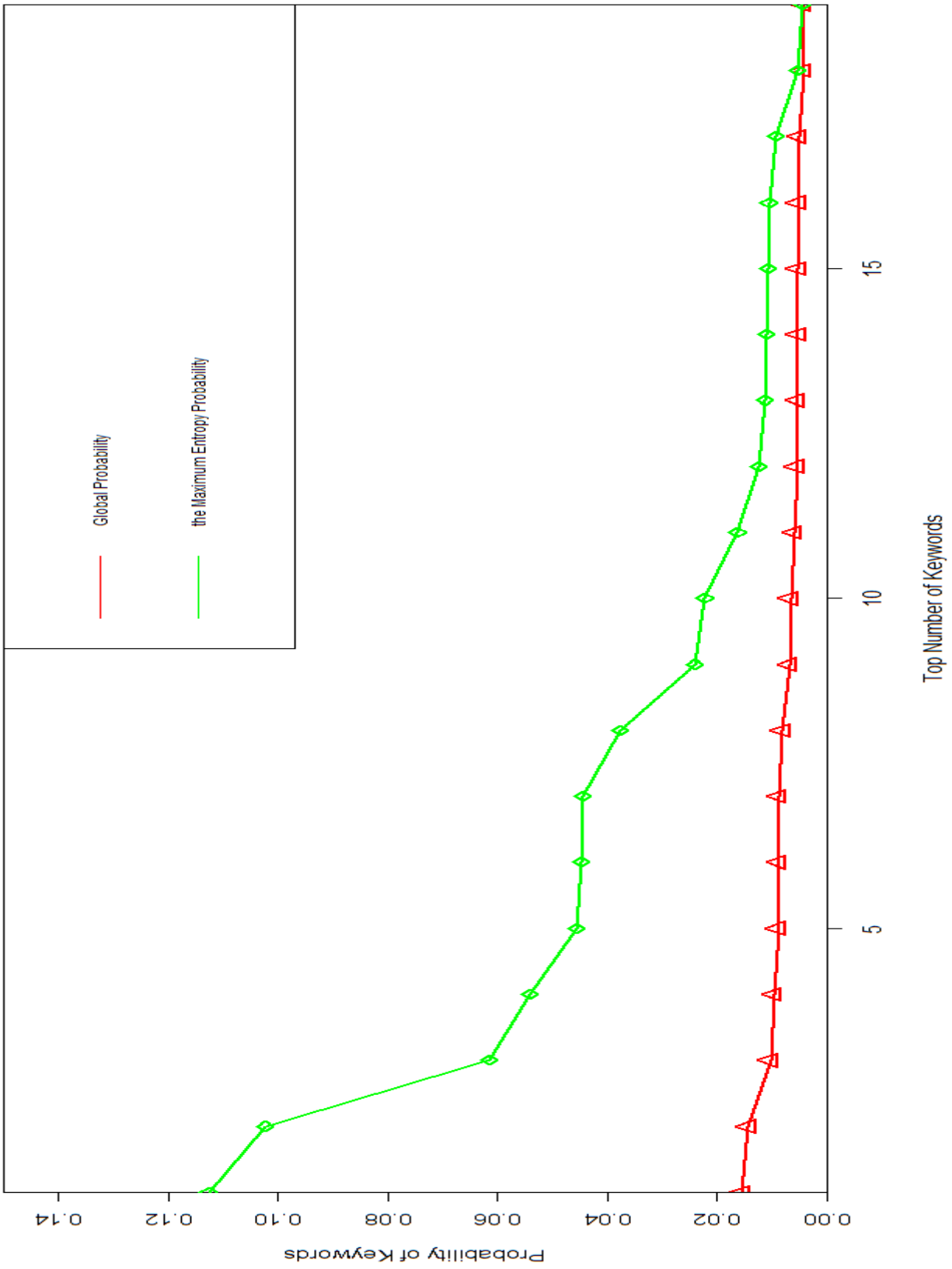


Figure 12: Relationship of Top 19 Keywords Probability

If we choose 21 words to be keywords, the relationship between global probabilities and maximum general entropy answers is:

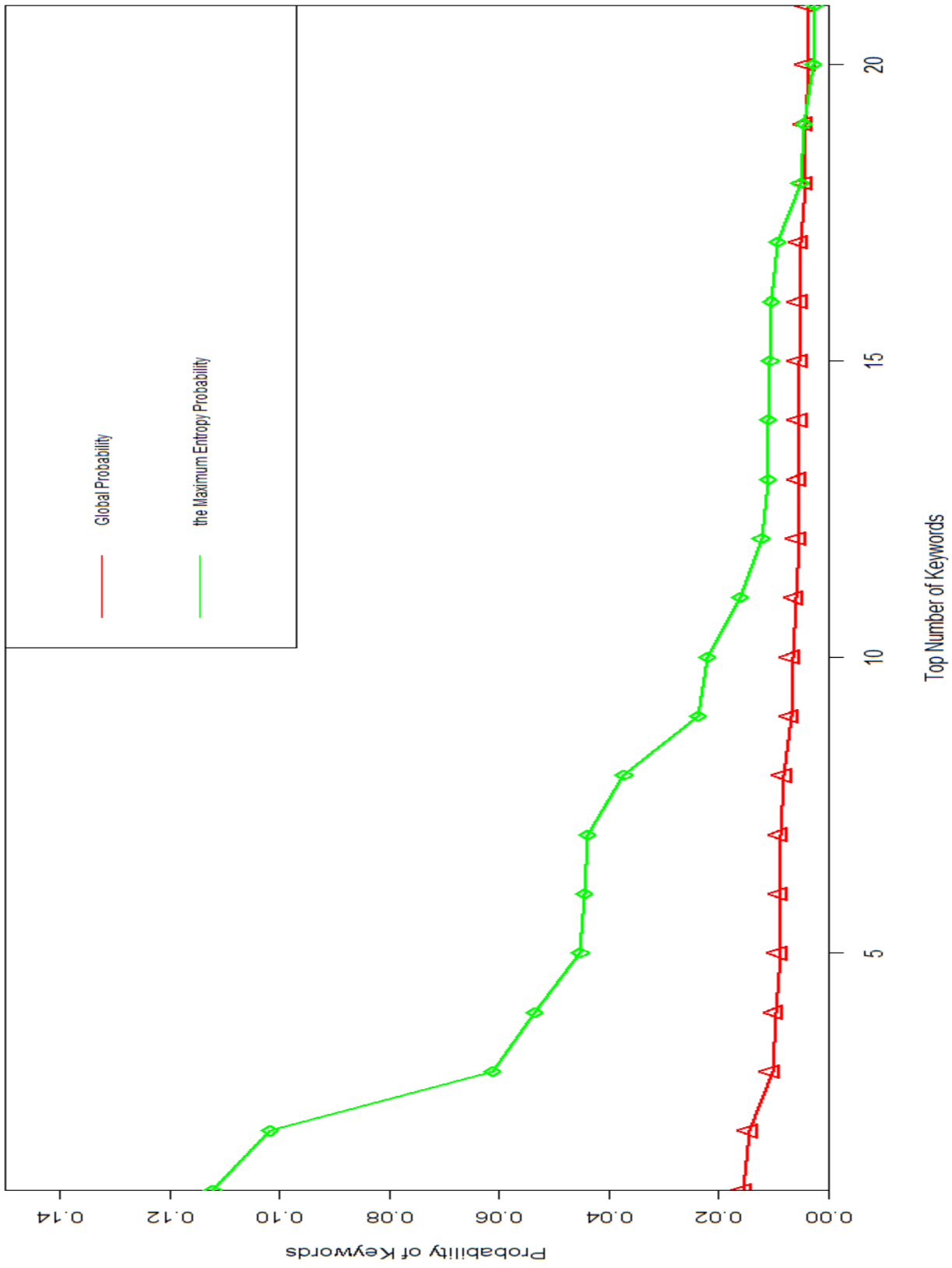


Figure 13: Relationship of Top 21 Keywords Probability

Figures 11, 12, and 13 clearly illustrate that though we initially select 100 keywords to analyze answers, the optimum number of keywords is actually 19. Thus, when we use a λ_0 determined by **Theorem 1**, we can calculate Q_L and Q_H in terms of (4-2). We plot them as follows:

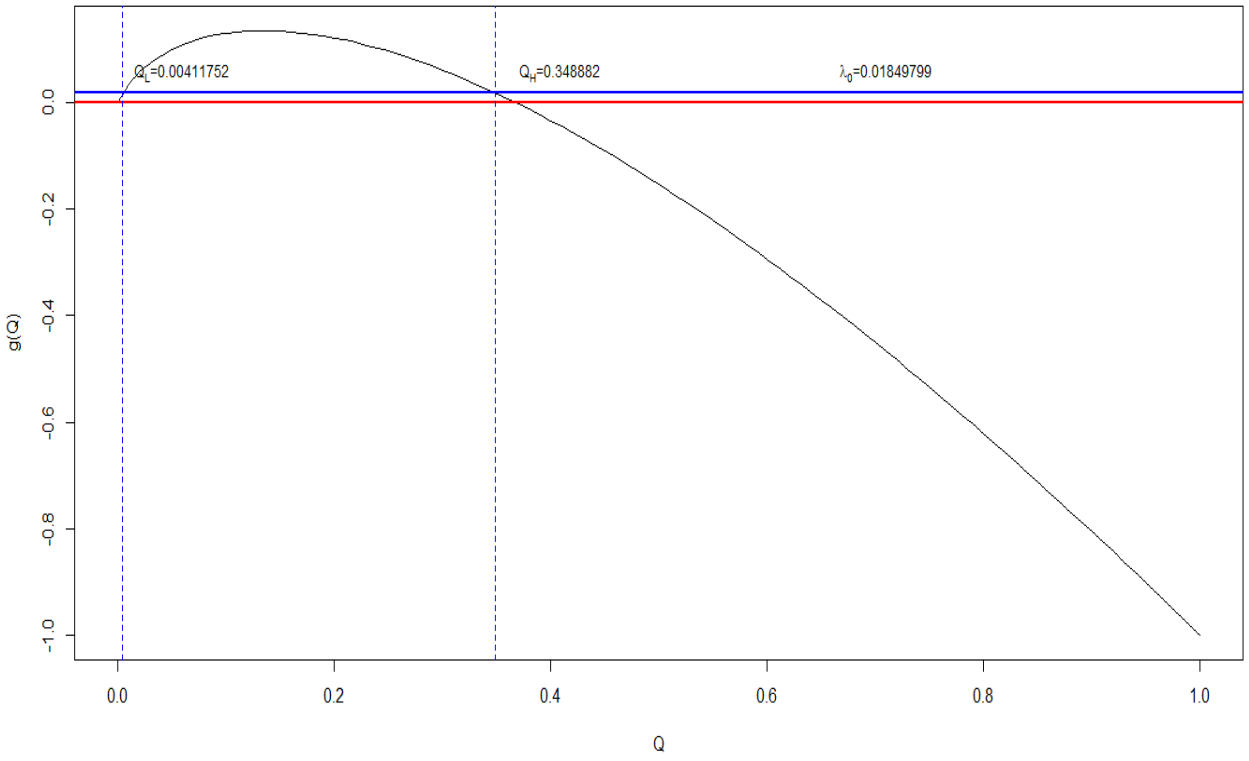


Figure 14: The cut-off value for Amazon Data

From Figure 14, we find $Q_L = 0.00411752$. Thus we throw away keywords with global probabilities smaller than Q_L. Global probabilities of 100 keywords are plotted as follows:

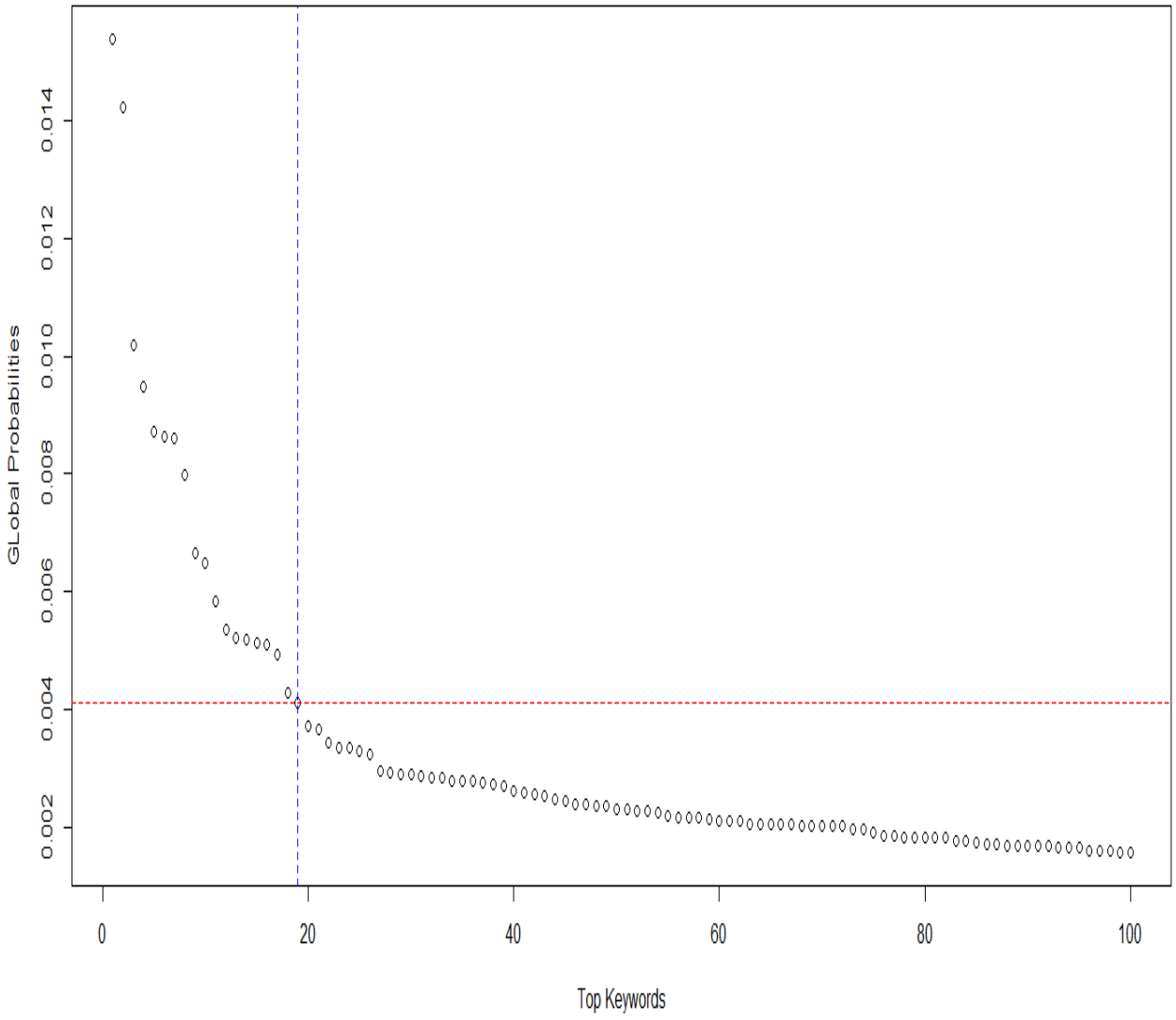


Figure 15: Top Number of kept Keywords

From Figure 15, it clearly shows that global probabilities of top 19 keywords are larger than Q_L, hence they will be kept and the other 81 keywords are discarded. We use (4-3) to obtain the relative efficiency of the kept keywords as follows:

P_{re}
93.16%

Though we throw away 81 keywords, we roughly throw away only 6.84% information from the viewpoint of the maximum general entropy answers. These dropped keywords should not affect the quality of the analysis.

Since we have known that $Q_i < B_i$ when $Q_L < Q_i < Q_H$, we can use the value of λ_0 in Figure 13 to show the relationship between B_i and Q_i when $0.00411752 < Q_i < 0.348882$:

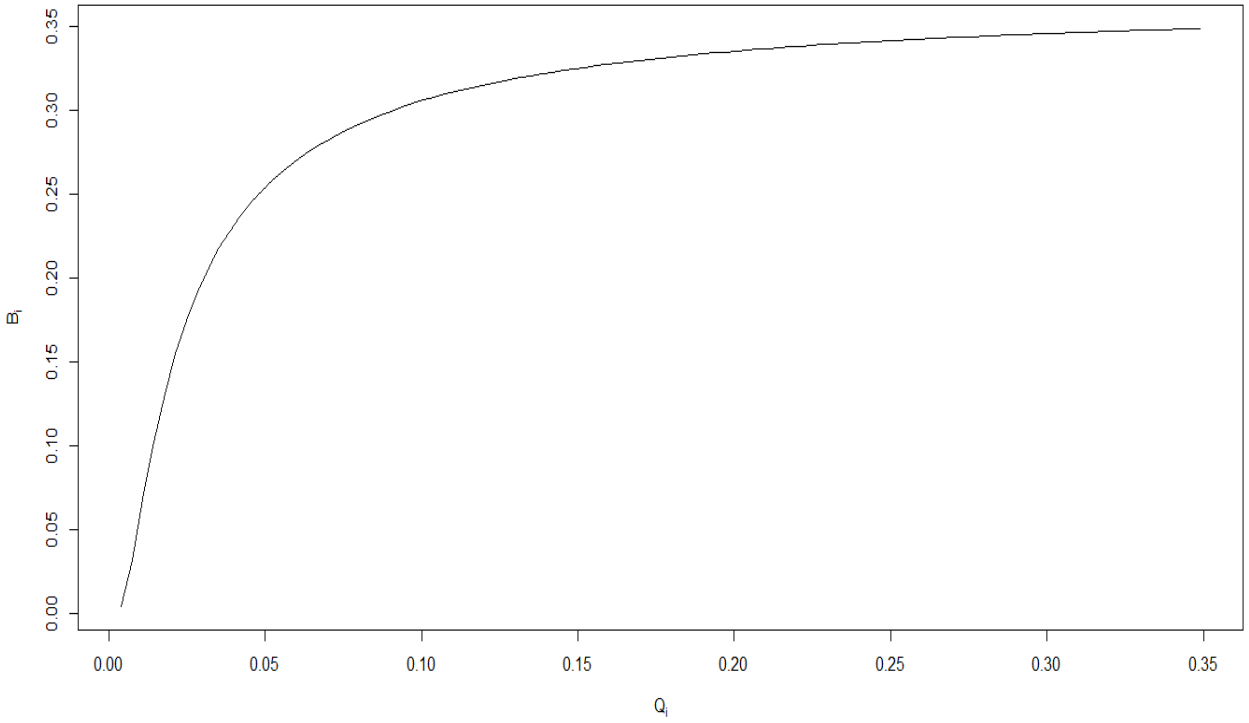


Figure 16: The relationship between the global probability and the maximum general entropy answer

Figure 16 clearly illustrates that there is a positive relationship between Q_i and B_i. The rate of B_i with respect to Q_i becomes small when Q_i is near Q_H.

To illustrate that the entropy of the best answer is better than the entropy of global answer, we plot their values for number of keywords and noise range from 0 to 19. Here, when we obtain Q_i and

$B_i, i = 0,1, \ldots, 19$, we try to compare contributions of $(-B_i \times Log(B_i))$ and $(-Q_i \times Log(Q_i))$ as follows:

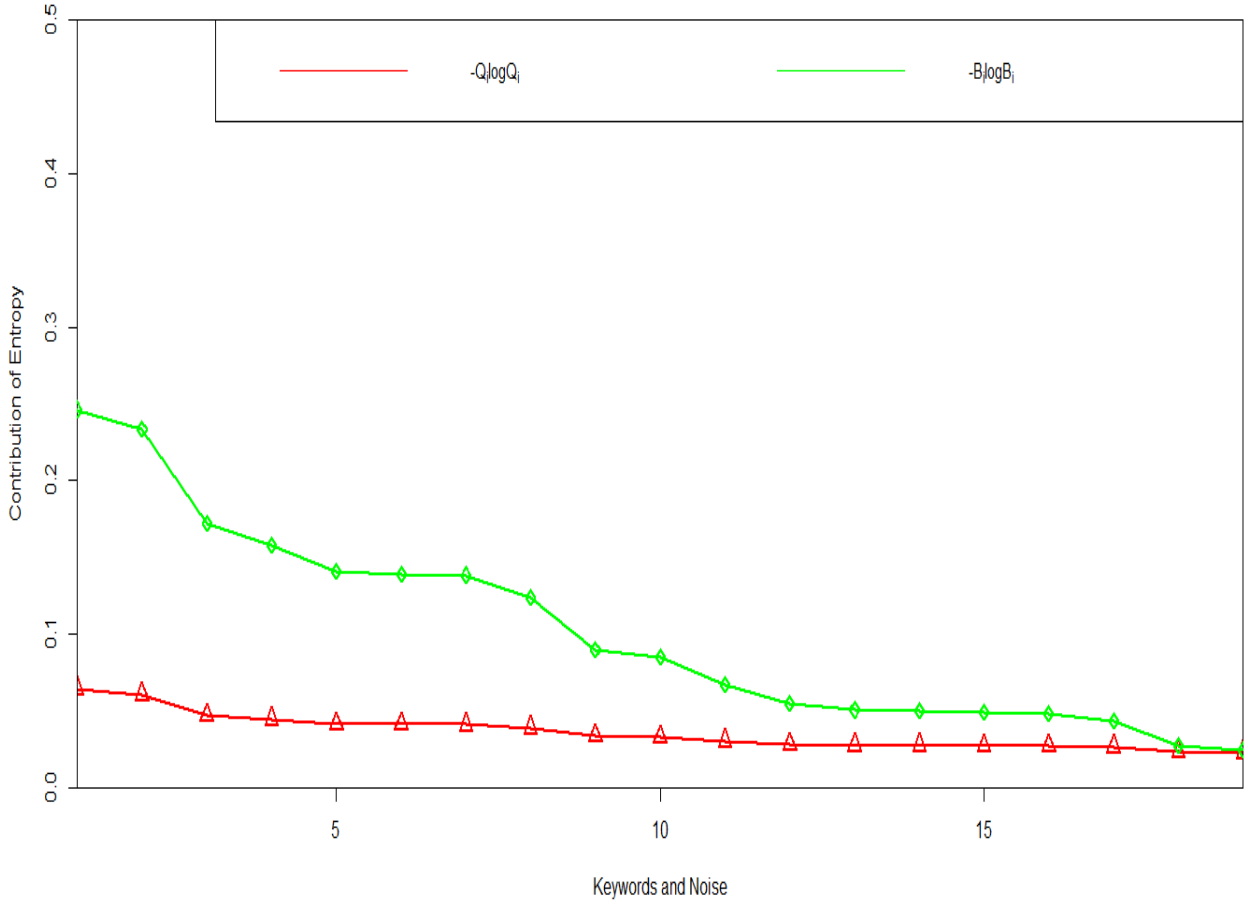


Figure 17: Relationships between the number of keywords and the contribution of the entropy

From Figure 17, we find that $-B_i \times Log(B_i) > -Q_i \times Log(Q_i), i = 0,1,2, \ldots, 19$. It means that contributions of $-B_i \times Log(B_i)$ are higher than contributions of $-Q_i \times Log(Q_i)$. It is clearly that $-Q_i \times Log(Q_i)$ and $-B_i \times Log(B_i)$ are all monotone increasing. However, $-B_i \times Log(B_i)$ is close to $-Q_i \times Log(Q_i)$ when the value of Q_i decreases.

We have shown the relationship between Q_i and B_i in terms of one value of λ_0. Now, we discuss relationships between Q_i and B_i in terms of different value of λ_0. We firstly discuss relationships

between Q_0 and B_0 in terms of different value of λ_0. Since the **Theorem 1** illustrates that the keywords number should be larger than two, we can use this Amazon example to show how B_0 variates in terms of different number of top keywords. For example, if the number of top keywords changes from 2 to 100, Q_0, B_0 and λ_0 respectively illustrate following variation:

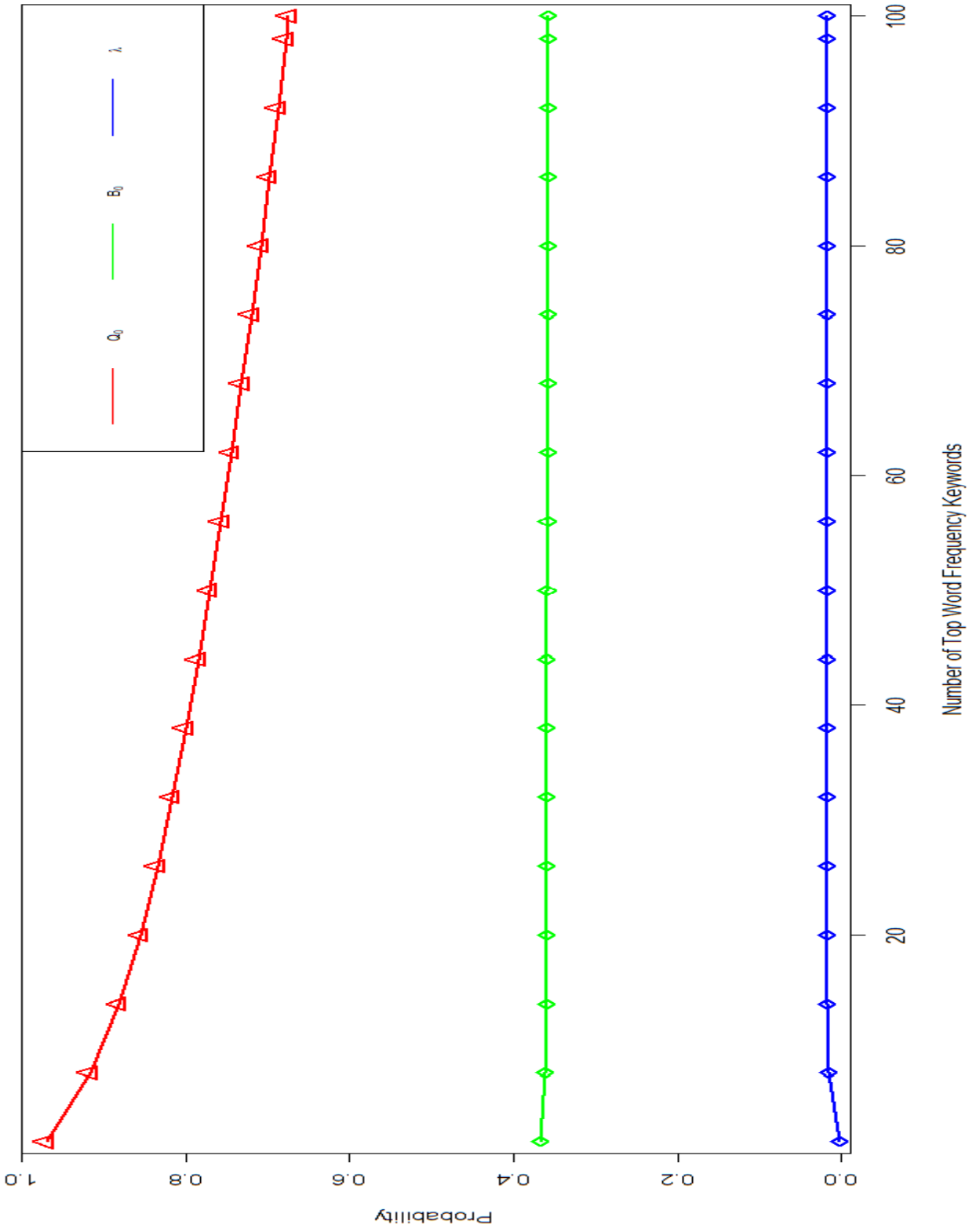


Figure 18: Relationship Between the Global Probability of noise and the Maximum General Entropy answer of noise in different parameters

From Figure 18, we find that, with respect to the number of keywords, λ_0 does not demonstrate an obvious change and Q_0 shows a clear change. Though B_0 shows a lightly monotone increasing when the number of top keywords is reduced, B_0 does not show an obvious change. Furthermore, when we choose different number of keywords, the noise content changes, which means that Q_0 may change. However, probabilities of top keywords do not change. Here, we do not discuss relationships among global probabilities of keywords, maximum general entropy answers of keywords and λ_0.

Figure 18 shows relationships between Q_0 and B_0 in terms of different λ_0. For the keyword i, we can also show Q_i and B_i in terms of different numbers of top keywords. Here, we choose some keywords from 19 keywords to analyze relationships among global probabilities, maximum general entropy answers, and λ_0. Relationships among Q_1, B_1, and λ_0 of some examples in answers are as follows:

Number of Top Keywords	Q_1	B_1	λ_0
5	0.015	0.169	0.012
7	0.015	0.141	0.015
11	0.015	0.122	0.017
13	0.015	0.119	0.017
15	0.015	0.116	0.018
18	0.015	0.113	0.018

From this table, we find that Q_1 does not change. Though λ_0 changes little, B_1 changes substantially. Similarly, global probabilities of other keywords do not change. Maximum general entropy answers of other keywords also change little.

When we select 19 keywords for analysis, we can plot the noise probability of each individual answer $\{P_0^1, P_0^2, \dots, P_0^{21405}\}$ as following:

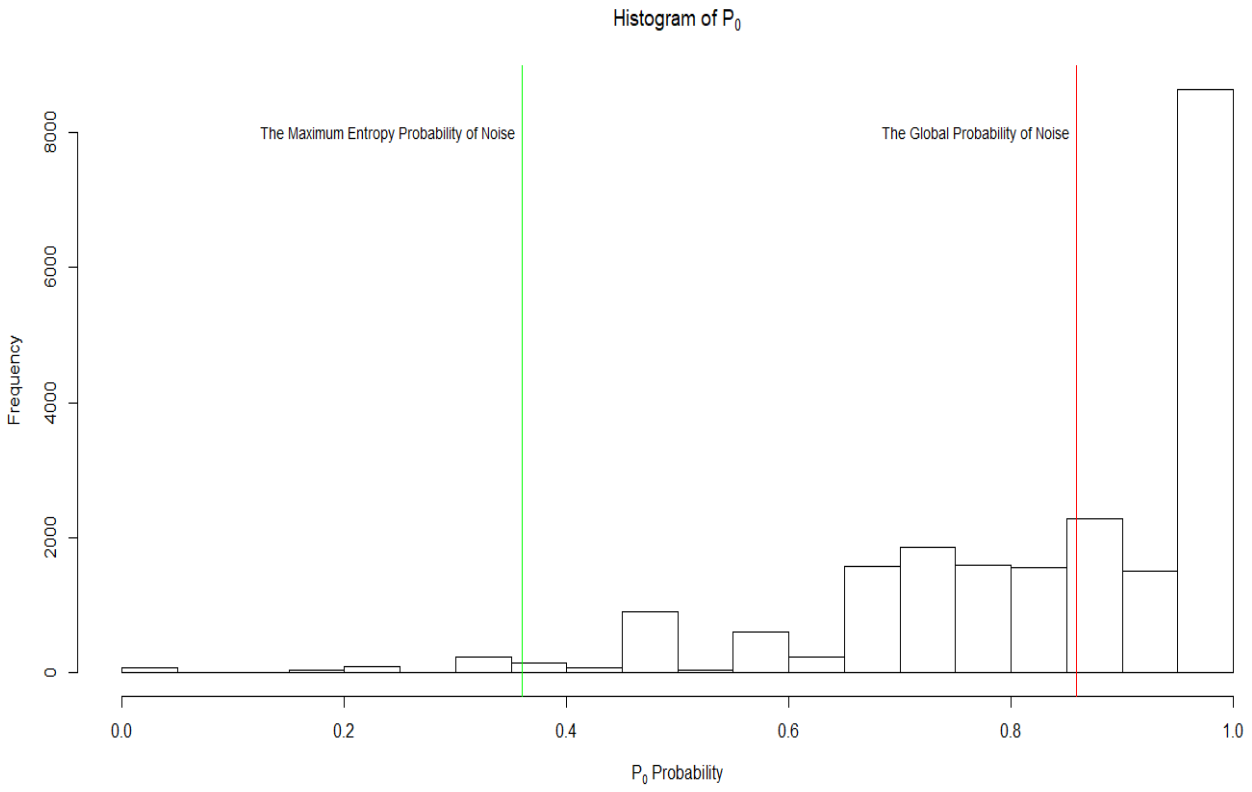


Figure 19: Histogram of Noise Probability

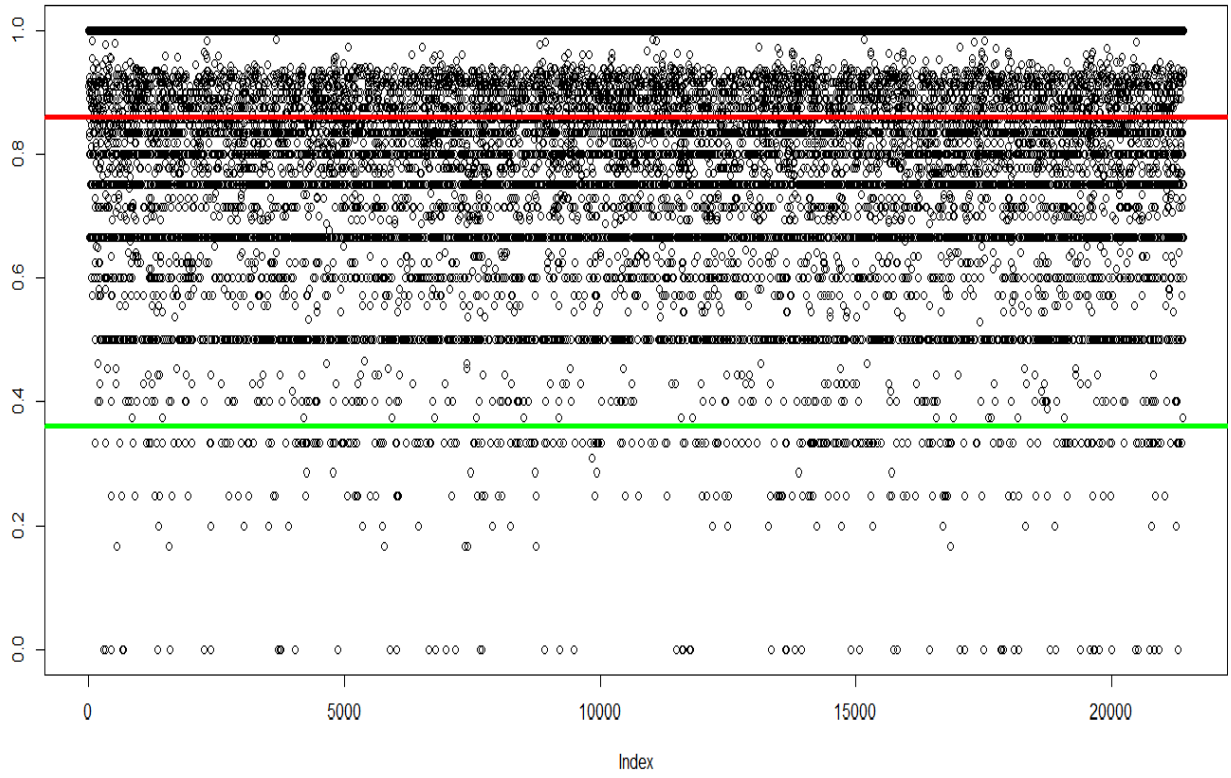


Figure 20: Distribution of Noise Probability

In Figures 19 and 20, noise probability of individual answers are represented by dots. The red line represents the global probability of noise: $Q_0 = 0.859$. The green line represents the maximum general entropy of noise: $B_0 = 0.360$. These two figures suggest that majority of an answer are noises. The minimum noise probability is 0.0 because all words in those answers are keywords. The maximum noise probability is 1.0 because all words in those answers are noises. Since we get Q_0 and B_0, we obtain percentages of different answers as follows:

$P_0 < B_0$	$B_0 < P_0 < Q_0$	$Q_0 < P_0$
1.94%	43.63%	54.43%

Table 10: Percentage of noise probability in different ranges

From Table 10, we find that roughly 54% of the answers are with $Q_0 < P_0$ and 43% are between B_0 and Q_0. Only about 2% of the answers are with $P_0 < B_0$.

Noise probabilities of individual answers reflect the quality of individual answers and probabilities of keywords. After analyzing noise probabilities, we want to analyze $E_n(\boldsymbol{P})$ for all these answers. Before we plot $E_n(\boldsymbol{P})$, we discuss some special situations:

(1) Sometimes, we may get a special answer, which contains equal numbers of keywords and noise. We can refer it as the uniform entropy. For example, if an answer contains one noise and one of each 19 different keywords, $E_n(\boldsymbol{P})$ of this answer is 0.1497866

(2) One answer may contain two keywords and no noise (e.g. the keyword 1 and the keyword 2), the general entropy of this answer is 0.01026087. We can refer it as the two-keyword entropy.

We also obtain the global entropy as 0.1182244. To compare the uniform entropy, the two-keyword entropy, and the global entropy, we plot them as follows:

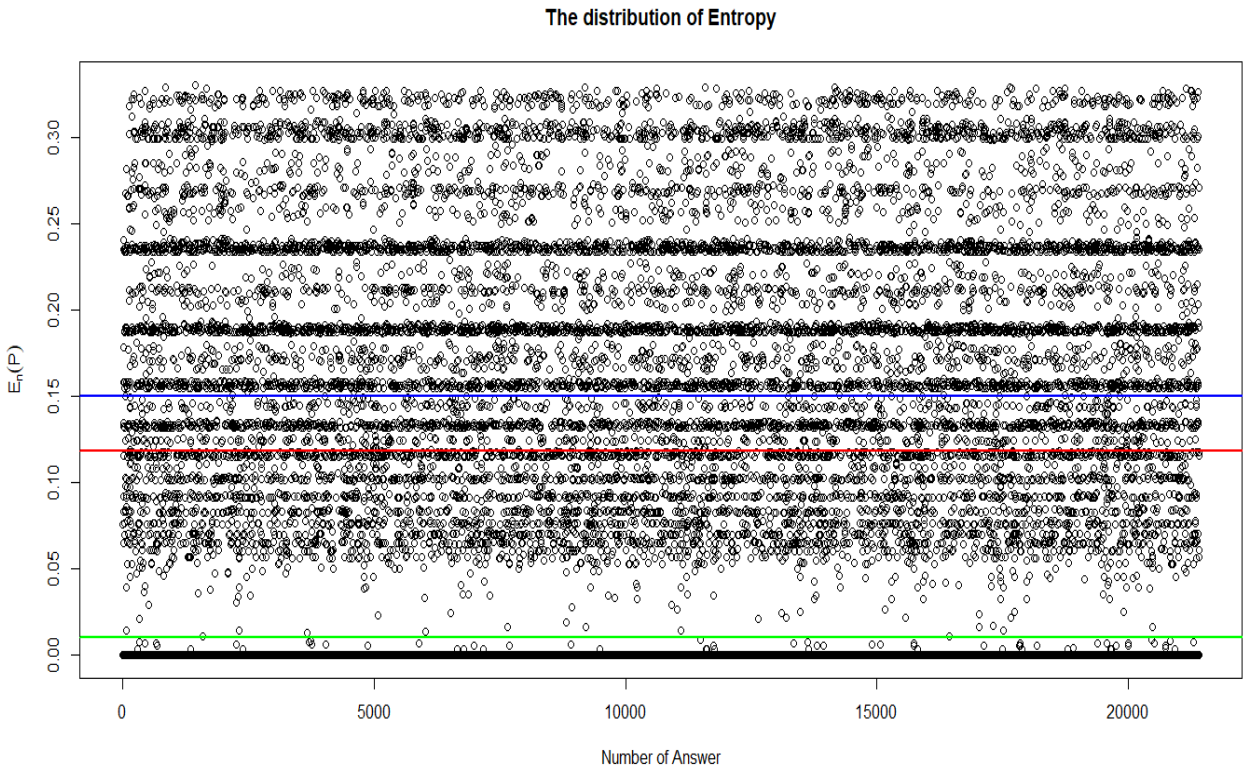


Figure 21: Distribution of General Entropy

In Figure 21, the general entropy of individual answers are represented by dots. The uniform entropy, the two-keyword entropy, and the global entropy are correspondingly represented by the blue, green, and red line. Summary of $E_n(\boldsymbol{P})$ is as follows:

Minimum of $E_n(\boldsymbol{P})$	Median of $E_n(\boldsymbol{P})$	Mean of $E_n(\boldsymbol{P})$	Maximum of $E_n(\boldsymbol{P})$
0.0	0.092	0.106	0.331

As a result, we find that the uniform entropy and the global entropy are all between mean value and maximum value of $E_n(\boldsymbol{P})$. The two-keyword entropy is smaller than the median value of $E_n(\boldsymbol{P})$. We introduce the uniform entropy and the two-keyword entropy because they can also be used to check fake answers, as value of the uniform entropy and the two-keyword entropy are not high value. If a fake answer is generated in terms of special chosen keywords, the value of entropy is also not too high. It means that the quality of this fake answer is not good. Additionally, we plot the histogram of $E_n(\boldsymbol{P})$ for answers as follows:

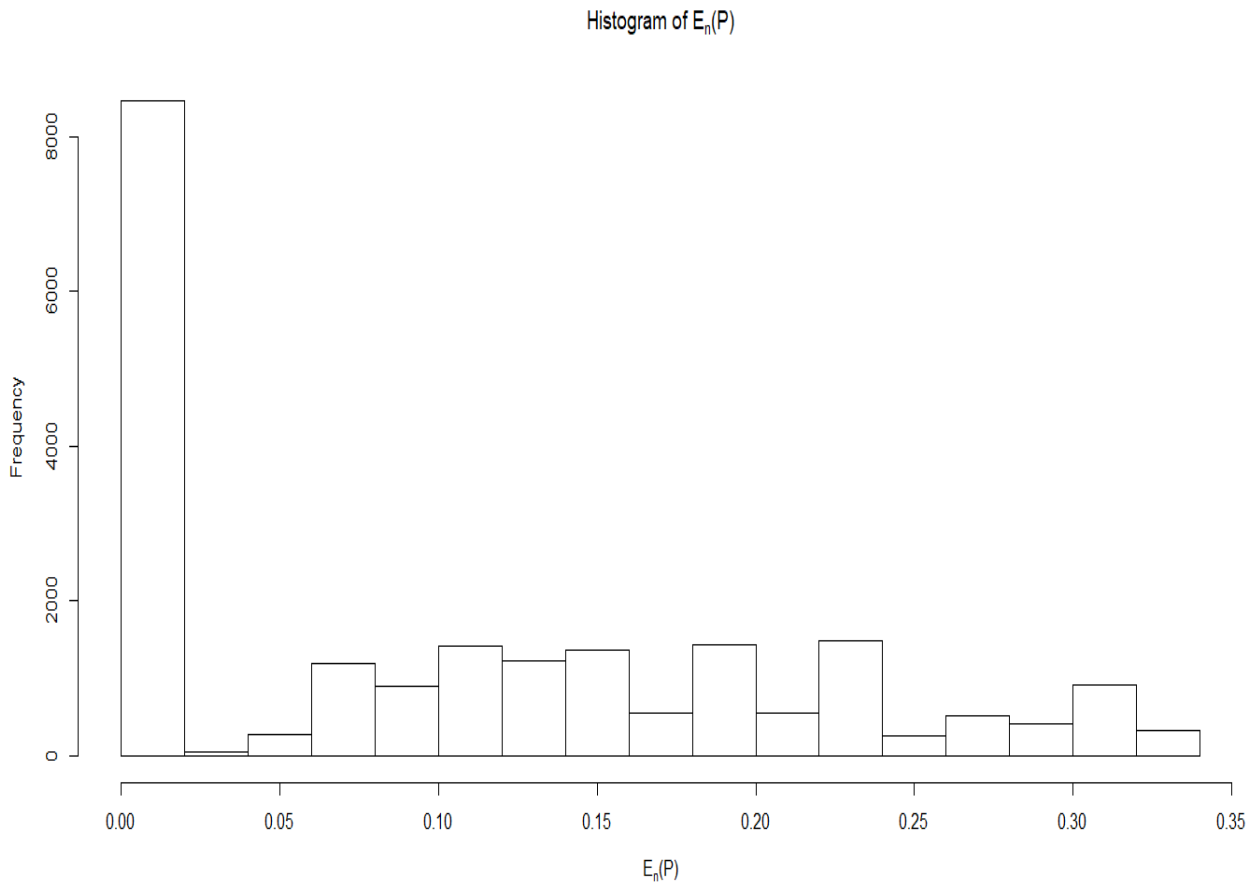


Figure 22: The histogram of the General Entropy

We can find two domains in this histogram figure. In one domain, $E_n(\boldsymbol{P})$ is 0. In the other domain, majority of $E_n(\boldsymbol{P})$ is between 0.05 and 0.35. Figure 22 clearly illustrates that the number of answers with $E_n(\boldsymbol{P})$ equals 0 is more than other answers.

Since we have developed CEW-DTW, we want to investigate into the relationships between the General Entropy and CEW-DTW. One answer may contain no keyword. The elements of such an answer are all zero when it is digitalized. Thus, this answer cannot be used to calculate Kullback Leibler distance with the ideal answer. Therefore, we select two answers with different CEW-DTW values, in which the first one is smaller than the second. We refer the first answer to be the Low-CEW-DTW answer and the second answer to be the High-CEW-DTW answer. When the 19 keywords are selected, we can calculate the global probability of noise Q_0 and global probabilities

of keywords $\{Q_1, Q_2, \dots, Q_{19}\}$. Then, we can get the maximum general entropy answer of noise B_0, and maximum general entropy answers of keywords $\{B_1, B_2, \dots, B_{19}\}$. We define the probability of Low-CEW-DTW answer to be $\{P_{0,CEW-DTW-L}, P_{1,CEW-DTW-L}, \dots, P_{19,CEW-DTW-L}\}$, where, $P_{0,CEW-DTW-L}$ is the noise probability and $P_{i,CEW-DTW-L}, i = 1,2, \dots,19$ is the keyword probability. Similarly, the probability of High-CEW-DTW answer is defined as $\{P_{0,CEW-DTW-H}, P_{1,CEW-DTW-H}, \dots, P_{19,CEW-DTW-H}\}$. We want to see whether the probability distribution of Low-CEW-DTW answer is closer to the maximum general entropy distribution than the probability distribution of High-CEW-DTW answer.

(1) Compare Q_0, B_0, $P_{0,CEW-DTW-L}$, and $P_{0,CEW-DTW-H}$

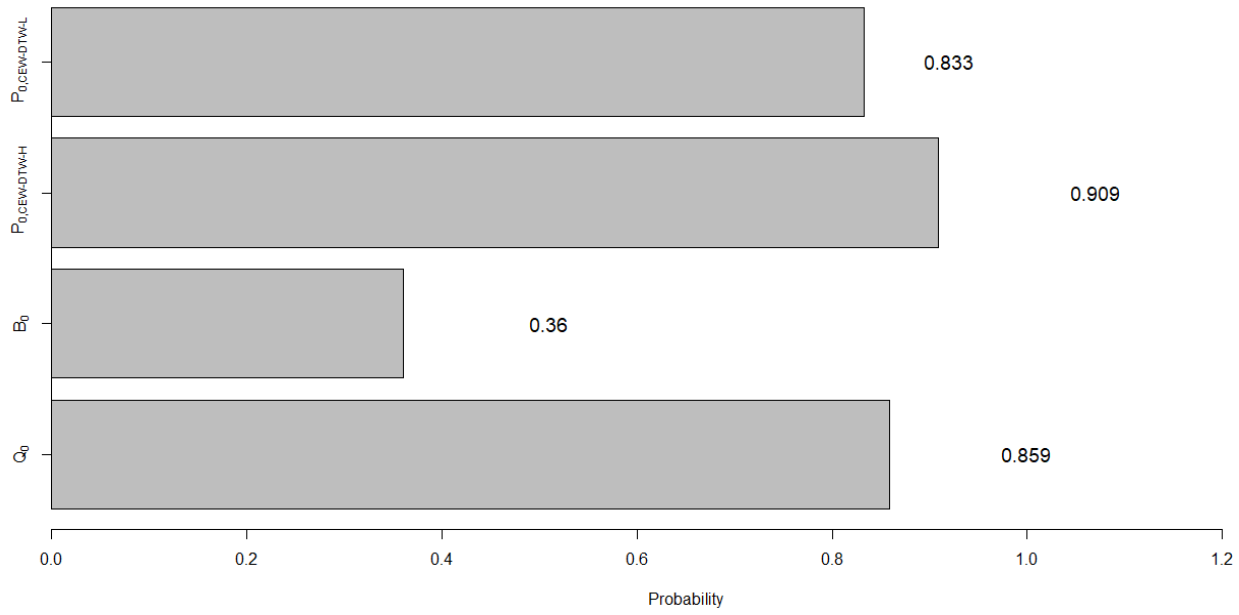


Figure 23: Noise Probability for Low or High CEW-DTW answers

Figure 23 suggests that $P_{0,CEW-DTW-L}$ is smaller than $P_{0,CEW-DTW-H}$. It means that the Low-CEW-DTW answer has less noise than the High-CEW-DTW answer. $P_{0,CEW-DTW-L}$ is closer to B_0 than $P_{0,CEW-DTW-H}$.

(2) Compare Q_i, B_i, $P_{i,CEW-DTW-L}$, and $P_{i,CEW-DTW-H}$

Keywords No.	Q_i	B_i	$P_{i,CEW-DTW-H}$	$P_{i,CEW-DTW-L}$
1	0.01538	0.11271	0.0	0.0
2	0.01422	0.10237	0.0	0.16667
3	0.01018	0.06157	0.0	0.0
4	0.00949	0.05406	0.0	0.0
5	0.00871	0.04556	0.0	0.0
6	0.00864	0.04473	0.0	0.0
7	0.0086	0.04434	0.0	0.0
8	0.00798	0.03757	0.0	0.0
9	0.00667	0.02405	0.05455	0.0
10	0.00648	0.02217	0.0	0.0
11	0.00583	0.01621	0.0	0.0
12	0.00537	0.01239	0.0	0.0
13	0.00521	0.01118	0.0	0.0
14	0.00518	0.01099	0.0	0.0
15	0.00514	0.01069	0.0	0.0
16	0.00512	0.01051	0.0	0.0
17	0.00494	0.00928	0.0	0.0
18	0.00427	0.00521	0.03636	0.0
19	0.00412	0.00445	0.0	0.0

Table 11: Keywords Probabilities in Different Answers

According to Table 11, most probabilities of keywords in the Low-CEW-DTW answer and the High-DTW answer are zero. Though some probabilities of keywords in the High-CEW-DTW

answer are higher than those in the Low-CEW-DTW answer, they do not affect the comparison result between the two types of answers.

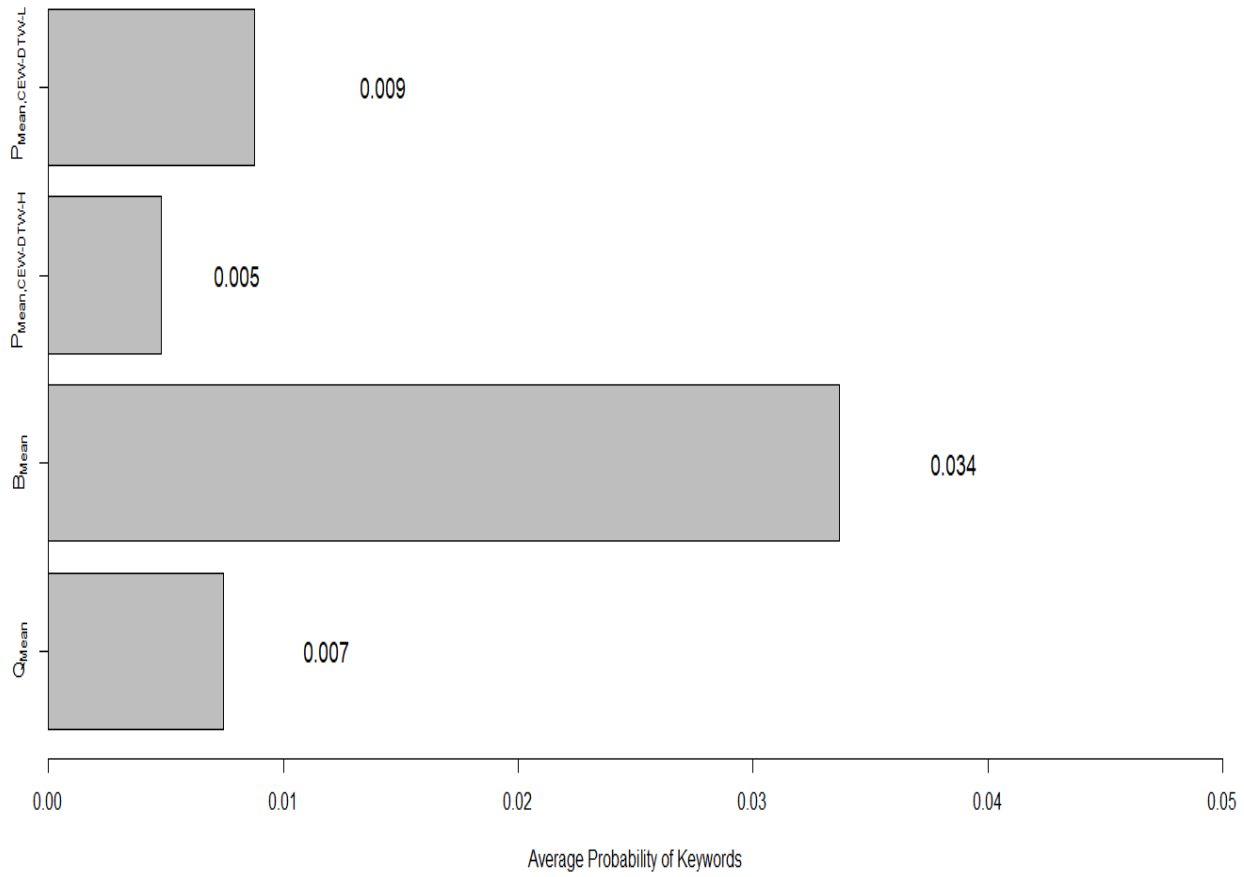


Figure 24: Mean Value of Keywords Probability for High or Low CEW-DTW answers

We use the mean value to better explain that the probabilities of keywords in the Low-CEW-DTW answer are closer to maximum general entropy answers than those in the High-CEW-DTW answer. In Figure 24, for the 19 keywords, the average value of global probabilities, the average value of maximum general entropy answers, the average value of probabilities in the Low-CEW-DTW answer, and the average value of probabilities in the High-CEW-DTW answer are respectively referred as Q_{Mean}, B_{Mean}, $P_{Mean,CEW-DTW-L}$ and $P_{Mean,CEW-DTW-H}$. The figure suggests that the average value of probabilities of Low-CEW-DTW answer is higher than that of the High-CEW-DTW answer, and is closer to B_{Mean} than that of the High-CEW-DTW answer.

If we analyze the sum of probabilities of keywords, we plot the analysis result as following:

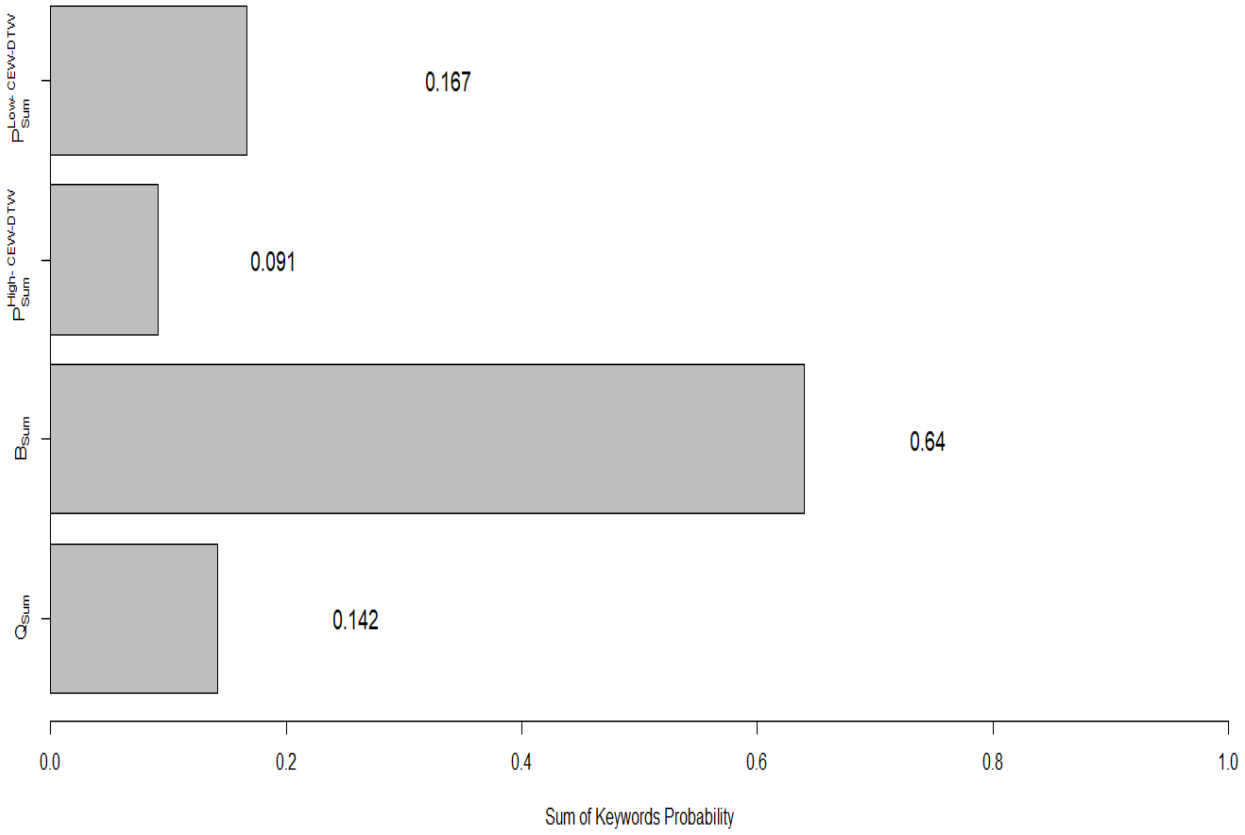


Figure 25: Sum of Keywords Probability for High or Low CEW-DTW answers

In Figure 25, the sum of the four quantities discussed above are respectively represented by Q_{Sum}, B_{Sum}, $P_{Sum,CEW-DTW-L}$, and $P_{Sum,CEW-DTW-H}$. This figure shows that the sum of probabilities of 19 keywords in the Low-CEW-DTW answer is higher than that of High-CEW-DTW answer, and is closer to the sum of maximum general entropy answers than that of High-CEW-DTW answer.

Since we define the global entropy, we can compare the global entropy, the general entropy of High-CEW-DTW answer and the general entropy of Low-CEW-DTW answer. We illustrate their difference as follows:

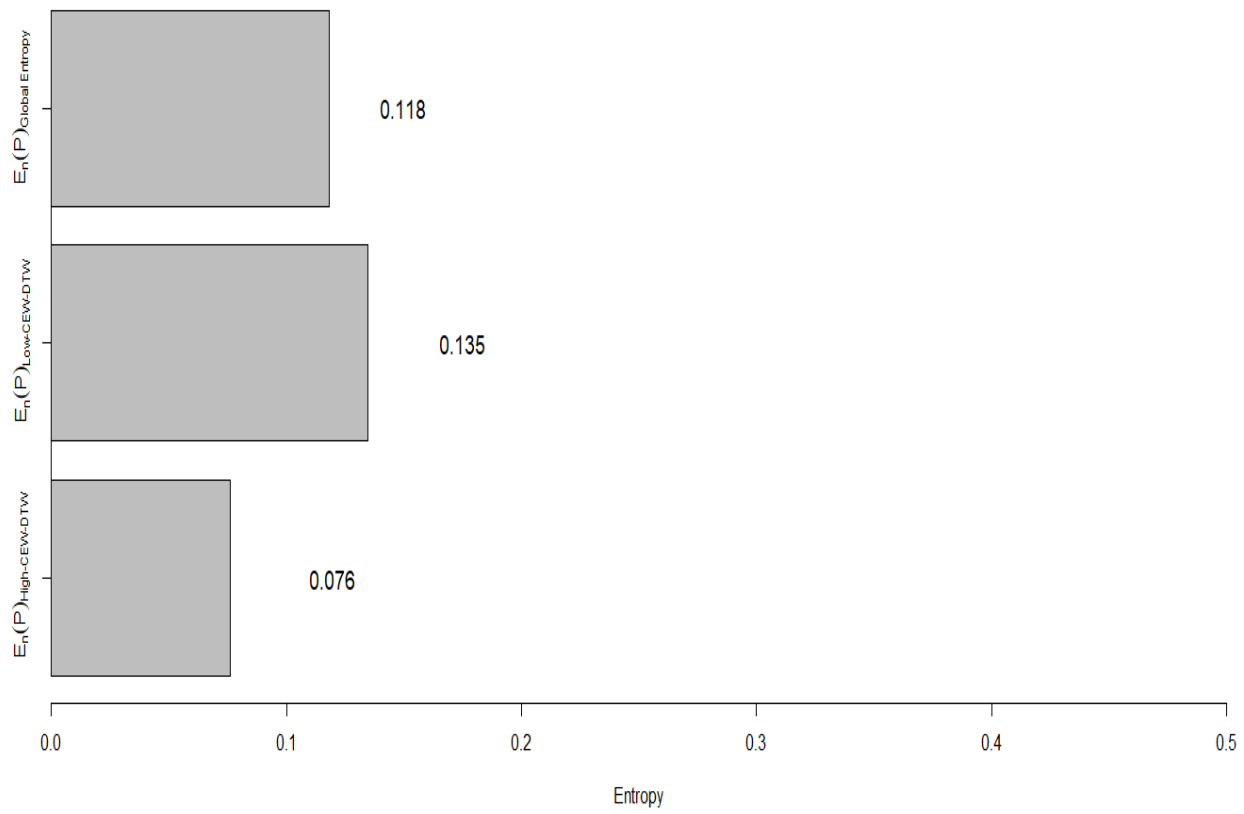


Figure 26: Comparison of the global entropy, the general entropy of High-CEW-DTW answer and the general entropy of Low-CEW-DTW answer

Figure 26 clearly illustrates that the global entropy is between the general entropy of High-CEW-DTW answer and the general entropy of Low-CEW-DTW answer. The Low-CEW-DTW answer has a higher general entropy than the High-CEW-DTW answer.

4.3.4 Apply the general entropy in a real Amazon product

Though we have used the general entropy to analyze Amazon answers, we want to see whether we can apply this methodology in a real scenario. Since Amazon also rank customers' comments for their products, we want to compare the general entropy and Amazon ranking to analyze their difference. We choose one product in Amazon (**Note:** Current link address is: https://www.amazon.ca/Evenflo-Tribute-Convertible-Seat-Bennett/dp/B0781ZBKP7/ref=cm_cr_arp_d_product_top?ie=UTF8). Though the percentage of comments with "5 stars" currently occupies 63%, comments with other stars are almost uniformly

distributed, and we can use such comments as examples for analysis. Figure 26 shows Amazon keywords of this product (**Note:** this picture in Figure 26 is a part of a webpage from the link above).

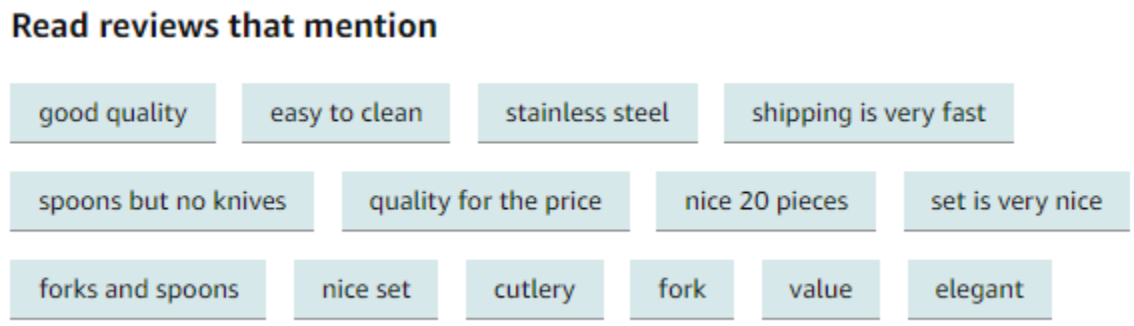


Figure 27: Comments of the product

Figure 28: Conditions of Sorting and Filtering

The Figure 28 shows conditions of sorting and filtering to rank comments (**Note:** this picture is a part of a webpage with current link: https://www.amazon.ca/Evenflo-Tribute-Convertible-Seat-Bennett/product-reviews/B0781ZBKP7/ref=cm_cr_dp_d_show_all_btm?ie=UTF8&reviewerType=all_reviews). When we obtain all ranked comments, we firstly remove stopping words. Then, we select comments with at least two words. Now, the total number of comments is 92 (**Note:** the number of comments may be adjusted if new customers make comments). In the next step, we calculate word frequencies for all words and choose some top frequencies words as keywords (e.g. top 30 as keywords). Then, we calculate global probabilities of these keywords. Also, we derive maximum entropy answers in terms of global probabilities. We plot these probabilities as follows:

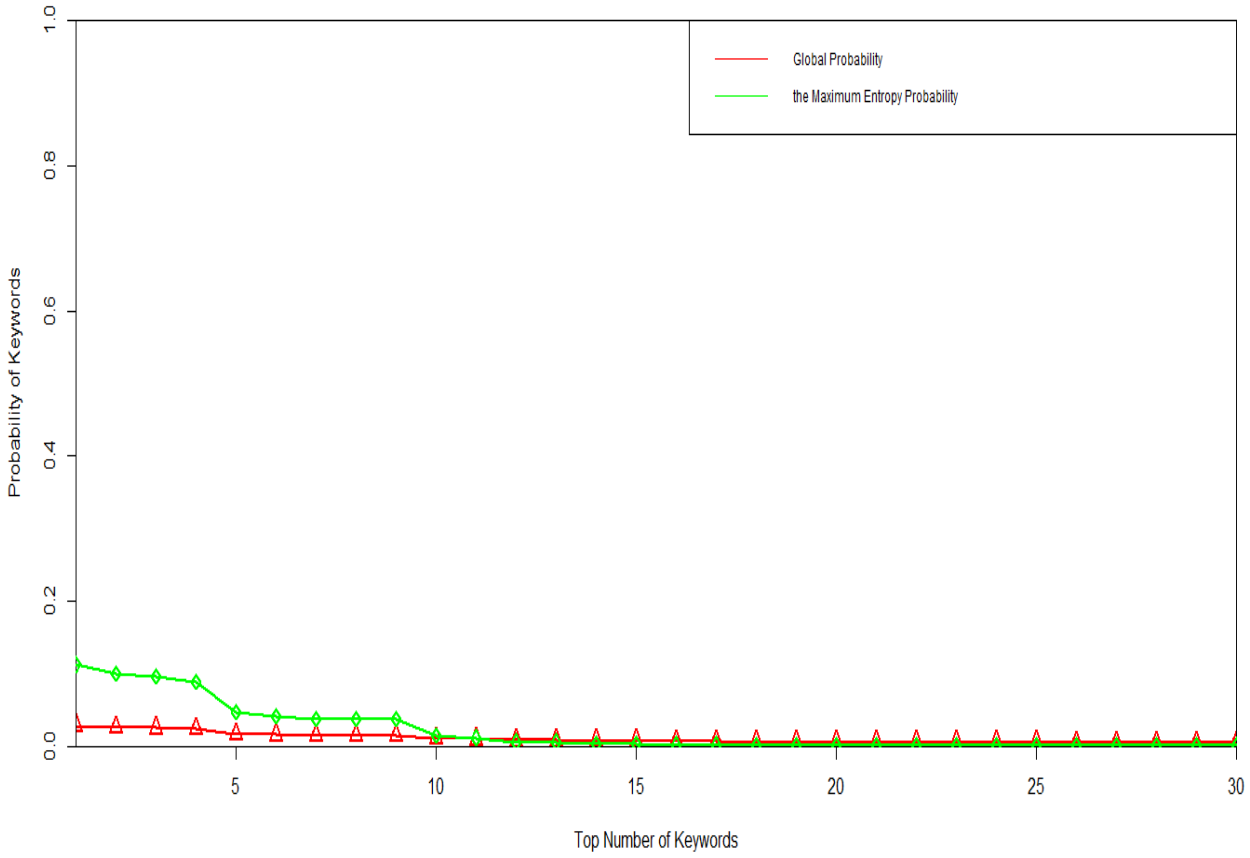


Figure 29: Relationship between Global probabilities and Maximum entropy answers

From Figure 29, we find that top 12 keywords are actually enough since maximum entropy probabilities of these keywords are higher than global probabilities. The following table is to compare keywords chosen by our methodology and Amazon keywords.

Top 12 Keywords	Amazon Keywords
quality, good, not, set, price, great, forks, like, nice, but, spoons, them	good quality, easy to clean, stainless steel, shipping is very fast, spoons but no knives, quality for the price, nice 20 pieces, set is very nice, forks and spoons, nice set, cutlery, fork, value, elegant

Table 12: Comparison of Keywords

From Table 12, we find that 8 keywords of our top 12 keywords (i.e. 66.67%) appear as Amazon keywords.

Now, we use these top 12 keywords to calculate $E_n(\boldsymbol{P})$ of comments and compare them with the ranking of Amazon comments.

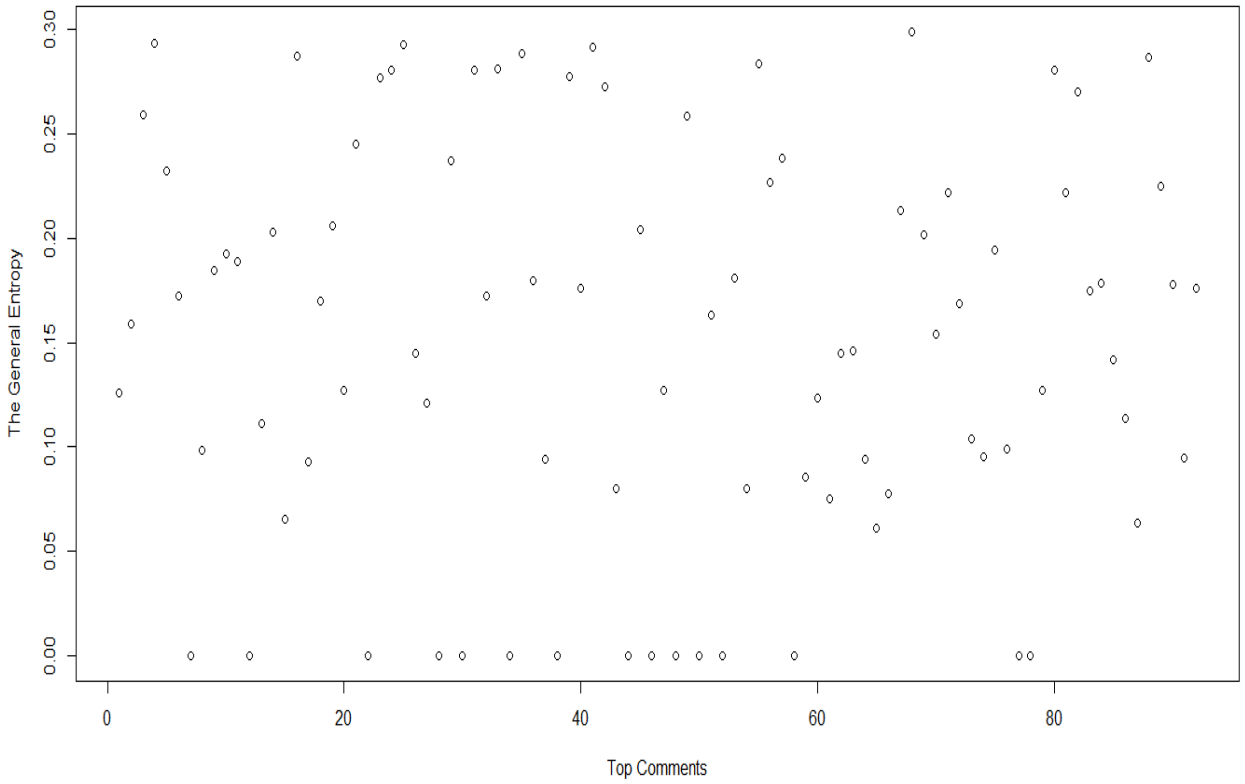


Figure 30: Comparison of the General Entropy and Ranking of Amazon comments

In Figure 30, the horizontal axis represents the ranking of top Amazon comments. The value 1 of the axis is the top one comment. The vertical axis represents the general entropies of these comments. The general entropies of comments have no too clear relationship with ranking of Amazon comments. For example, the value of the general entropy of No.88 comment is quite large, but this comment is not located at a better place in the queue of comments by Amazon ranking. Similarly, the value of the general entropy of No.7 comment is the minimum value, but this comment is located before No.88 comment (**Note:** Currently, the rank numbers of these

comments are 88 and 7 respectively, but they may change in the future if new comments are added). The contents of these comments are shown in the following table:

No.	Content of Comments
88	solid utensils great quality great value
7	initially shipped item defected seller send replacement much better receive 5 stars receive replacement item initial shipment one star taken extra time spend get final perfect item

Table 13: Comments for Products in Amazon

When we use top 12 keywords to analyze comments, we find that three of the six words are keywords in No.88 comment, hence this comment has a high value of $E_n(\boldsymbol{P})$. On the other hand, No.7 comment contains no keywords, so $E_n(\boldsymbol{P})$ of this comment is small. Thus, if we use the general entropy to rank comments, the No.88 comment should be located before the No.7 comment. The general entropy ranks comments in terms of probabilities of keywords and noises. If a comment contains more keywords and less noise, the quality of this comment is good. The general entropy does not judge the meaning of words. Another reason for the general entropy of comments to be small is that some comments are written in French (e.g. No.22 comment). So, they will be regarded as noises. However, though a comment may not contain any keyword, Amazon may adapt other methodologies to prove the content of this comment to be a high-quality one. Therefore, it is also reasonable to make such a comment rank high in Amazon comments. Though the general entropy and Amazon ranking show certain differences in ranking, they elaborate ranking from different viewpoints respectively. Therefore, we think these methodologies are both reasonable.

4.4 Survey

4.4.1 Purpose of Survey

The purpose of this survey is to assess which methodology is closer to the subjective judgment of human by comparing CEW-DTW and the General Entropy.

4.4.2 How to design survey

Since readers cannot remember too many keywords when reading, we define five keywords for this survey. The length of answers will also affect the accuracy of judgement. We pick up some answers and let readers to use these keywords to judge qualities of these answers. Since CEW-DTW and the General Entropy can be applied to assess qualities of answers, our purpose of this survey is to see which methodology of them is closer to subjective assessment. However, these two methodologies do not care about the actual meaning of keywords or sentences. Therefore, we require readers to focus on keywords and sentences themselves. It is not necessary for readers to care about the actual meaning of keywords and sentences. The survey example is as follows:

The survey objective: This survey is to assess how following texts are relevant to **keywords**. (**Don't care about the actual meaning of keywords and sentences.)**

Test Keywords: **"rear", "power", "system", "roof", "store"**

Texts Assessment (Rank Score: {3,2,1})

3: HighText Quality

2: Medium Text Quality

1: Low Text Quality

- It should work with most power wheels

  Rank Score:

- No for a rear Jeep seatmines on a 99 wranglerlove it

  Rank Score:

- Plastic

  Rank Score:

- Installed weight on this system is about 37 pounds Shipping is about 40 pounds

  Rank Score:

- these are power adjusting manual folding and heated

  Rank Score:

- As clear as I have seen Obviously it adds gloss it looks similar to clear bra when applied

  Rank Score:

Figure 31: The Survey Example

We randomly select ten people on campus to do this survey. These people are provided six answers and asked to use three labels (e.g. “High Text Quality”, “Medium Text Quality”, and “Low Text Quality”) to assess these answers. Some survey results as follows:

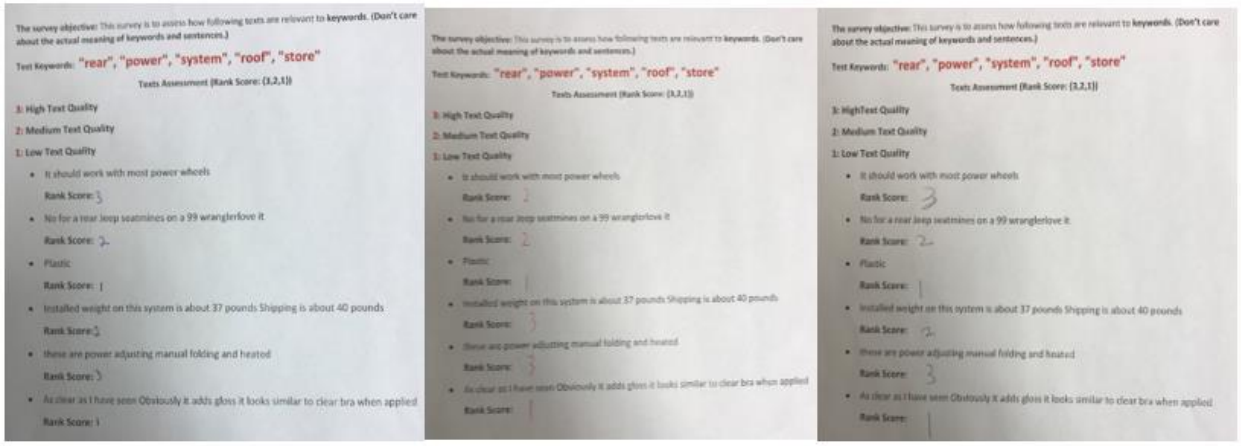


Figure 32: Survey Examples

4.4.3 Analysis and Comparison

4.4.3.1 Analysis by CEW-DTW and the General Entropy

We firstly use these keywords to digitalize survey answers. Then, we generate an “ideal” answer, with length equal to the maximum length of these answers. We use this “ideal” answer to calculate CEW-DTW as follows:

Answer No	Answers	CEW-DTW
1	Plastic	4.773
2	As clear as I have seen Obviously it adds gloss it looks similar to clear bra when applied	9.281

3	No for a rear Jeep seatmines on a 99 wranglerlove it	19.467
4	Installed weight on this system is about 37 pounds Shipping is about 40 pounds	25.306
5	these are power adjusting manual folding and heated	13.627
6	It should work with most power wheels	11.68

Table 14: Value of CEW-DTW in Survey Examples

The rule of CEW-DTW is that if a vector is closer to the "ideal" vector, this vector has smaller CEW-DTW value. This table illustrates that Answer 1 and Answer 2 are the most close to the "ideal" answer than other answers. Answer 5 and Answer 6 are closer to the "ideal" answer than Answer 3 and Answer 4. We can regard the group of Answer 1 and Answer 2 to be in the first group since they have highest answer qualities. Similarly, the group of Answer 5 and Answer 6 are in the second group since they have medium answer qualities. The group of Answer 3 and Answer 4 are in the third group since they have lowest answer qualities.

Answer No.	**Answers**	**Rank By CEW-DTW**
1	Plastic	the first group
2	As clear as I have seen Obviously it adds gloss it looks similar to clear bra when applied	

3	No for a rear Jeep seatmines on a 99 wranglerlove it	the third group
4	Installed weight on this system is about 37 pounds Shipping is about 40 pounds	
5	these are power adjusting manual folding and heated	the second group
6	It should work with most power wheels	

Table 15: Group Categories of Survey Examples

However, the group of Answer 1 and Answer 2 have no keywords. Though, Answer 3, Answer 4, Answer 5 and Answer 6 contain one keyword respectively, length of the group of Answer 6 and Answer 5 are obviously shorter than others respectively. It means that the group of Answer 5 and Answer 6 has higher keywords ratio than the group of Answer 3 and Answer 4. Thus, our goal is presented that the group of Answer 5 and Answer 6 should be in the first group; the group of Answer 3 and Answer 4 should be in the second group; the group of Answer 1 and Answer 2 should be in the third group.

When we want to calculate the general entropy of these answers, we firstly rank these keywords and calculate their global probabilities. Then, we calculate individual probabilities of these keywords in each individual answer. Results of the General Entropy are as follows:

Answer No	**Answers**	$E_n(P)$
1	Plastic	0.0

2	As clear as I have seen Obviously it adds gloss it looks similar to clear bra when applied	0.0
3	No for a rear Jeep seatmines on a 99 wranglerlove it	0.087
4	Installed weight on this system is about 37 pounds Shipping is about 40 pounds	0.069
5	these are power adjusting manual folding and heated	0.117
6	It should work with most power wheels	0.132

Table 16: The value of the General Entropy value in survey examples

Entropy reflects the information ratio of keywords in answers. If the entropy of one answer is high, this answer is considered as to have a high quality.

4.4.3.2 Survey Conclusion

By comparing survey results with CEW-DTW and the general entropy, we find that the survey result matches the result of general entropy more relatively than the result of CEW-DTW. Therefore, we conclude that the general entropy test is more reasonable than CEW-DTW.

4.5 Conclusion

In this chapter, we develop the general entropy method to analyze answers, which combines the noise probability and keywords probability together to test qualities of answers. We use the general

entropy to further derive the maximum entropy answer, which can be considered as an optional goal to judge actual answers. Compared with the “ideal” answer in CEW-DTW, the maximum entropy answer can find a group of answers, which can be considered to have high qualities. It agrees with the actual practice that we cannot always regard an answer as the best one in a group of answers. The maximum general entropy answer also gives us a way to find the number of optimum keywords. In our real scenario analysis, the number of optimum keywords is approximately equal to the number of keywords for real Amazon products. We also find that the methodology of the general entropy approximately agrees with the methodology of CEW-DTW for checking qualities of answers. By applying the general entropy to analyze comments of a real Amazon product, we obtain the number of optimum keywords. Some of these keywords also appear in the Amazon keywords. We organize a small survey to assess some answers. Compared with the assessment results by CEW-DTW, we find that the survey results agree more with the assessment results by the general entropy. It means, to some extent, the general entropy is more in line with the reality than CEW-DTW.

The methodology of the general entropy can be applied in many ways. In addition to assess the qualities of answers or comments and find the number of optimum keywords, we can use the general entropy to filter documents (e.g. Curriculum Vitae). This methodology will help people to find qualified documents in terms of keywords. Also, we can apply this methodology in marketing research. It contributes to find market heats. Additionally, we can also apply this methodology to verify qualities of voice with noise.

Chapter 5

5 Background of Markov Entropy

The trend in keyword sense is a technique that is used to automatically judge the intended transition between keywords in written documents. If a person makes a presentation about some topic without the limitation of time, his or her speech will usually focus on some keywords in terms of personal experience or knowledge base. Identifying the keywords in a person's speech will allow the reader to determine the main intuition of the speaker. Here, we define the keywords' trend to be the inner connection of keywords. We use the transition from one keyword to another to describe such an inner connection. Most transitions happen between a keyword and noise or between noise and noise, we also analyze these transitions. Currently, we analyze the inner connection between two contiguous words. This analysis can be used to determine which word connections are important in an answer.

Methodologies developed in Chapters 2 to 4 analyze texts based on the frequency and semantic distance of individual keywords rather than the connections between words. In this chapter we extend this idea by introducing a new methodology to incorporate the connections between keywords. We achieve this with a Markov transition process that models the transition from each word to the next.

5.1 Markov Transition Process

5.1.1 Literature Review

Markov approaches have been applied in many researches. Bennett and Hauser [112] use Markov-based approaches to develop an Artificial Intelligence simulation framework, which can be used for automatic decision support in a clinical framework. Tiomkin and Tishby [113] also use Markov technology to develop methodologies to model bi-directional interactions between organisms and their environments, where the organisms maximize their rewards via Markov decision processes. Pollock et al. [114] obtain a necessary and sufficient condition for a quantum process to be

Markovian which is asymptotically equivalent to the classical limitation but provides additional methods for determining non-Markovianity. Other references, e.g. Kang et al. [115], George et al. [116] have used Markov chains for text mining, but their methods are mainly based on hidden Markov models rather than keyword or noise transitions.

5.1.2 Transition Matrix

We begin by selecting some typical keywords then calculate transition probabilities from one word to another. We can find a transition matrix **C** for transitions of all words as follows:

$$\begin{pmatrix} C[0][0] & \cdots & C[0][n] \\ \vdots & \ddots & \vdots \\ C[n][0] & \cdots & C[n][n] \end{pmatrix},$$

where, $i = 0,1,2,\dots,n$ and $j = 0,1,2,\dots,n$. $C[i][j]$ represent the transition frequency from the word i to the word j. $\sum_{j=0}^{n} C[i][j] = 1, i = 0,1,2,3,\dots,n$. $C[0][0]$ represents the transition frequency from noise to another noise; $C[i][0]$ represents the transition frequency from a keyword to noise; $C[0][i]$ represents the transition frequency from noise to a keyword. We do not distinguish between noise words; all noise words are treated as if they are the same and are indexed by 0. We call this matrix as **the Transition Matrix.**

5.2 Markov Transition Probability Model

5.2.1 Introduction of Markov Transition Matrix

The transition matrix allows us to calculate the probability of any transition. When we select n keywords from a group of answers, we can obtain a probability matrix of transitions for all keywords and noise. We call this probability matrix to be the **global transition probability matrix**, it is:

$$\begin{pmatrix} Q_{0,0} & \cdots & Q_{0,n} \\ \vdots & \ddots & \vdots \\ Q_{n,0} & \cdots & Q_{n,n} \end{pmatrix},$$

where, $Q_{0,0}$ is the global transition probability from one noise to another noise; $Q_{0,i}, i = 1,2, \dots, n.$ is the global transition probability from noise to a keyword; $Q_{i,0}, i = 1,2, \dots, n.$ is the global transition probability from a keyword to noise; $Q_{i,j}, i, j = 1,2, \dots, n$ is the transition probability from one keyword to another keyword. $\sum_{j=0}^{n} Q_{i,j} = 1, i = 0,1,2,3, \dots, n.$ The transition probability for noise to keywords, noise to noise, and keyword to noise is usually larger than 0, but there are some pairs of keywords with zero probability of transition. In the following, we only consider the situation that each row has at least three non-zero transition probabilities. Similar to the global probability matrix, we can also obtain the transition probability matrix for each answer rather than for all answers. For an answer, the matrix is:

$$\begin{pmatrix} P_{0,0} & \cdots & P_{0,n} \\ \vdots & \ddots & \vdots \\ P_{n,0} & \cdots & P_{n,n} \end{pmatrix},$$

where, $P_{0,0}$ is the transition probability from one noise to another noise; $P_{0,i}, i = 1,2, \dots, n.$ is the transition probability from one noise to one keyword; $P_{i,0}, i = 1,2, \dots, n.$ is the transition probability from one keyword to one noise; $P_{i,j}, i, j = 1,2, \dots, n$ is the transition probability from one keyword to another keyword. $\sum_{j=0}^{n} P_{i,j} = 1, i = 0,1,2,3, \dots, n.$ This matrix is also called the **individual transition probability matrix**.

5.2.2 An example about Transition Probabilities

Using these rules for creating global and individual transition probabilities, we give an example to show the process. When we get a sentence, we remove stopping words and collinearity (described in detail in Chapter 6). For example, some keywords and a group of answers are:

Top Keywords:	seat, fit, baby, use
Answer 1	gate plus extension fit well inch opening concerned max fit well
Answer 2	wide base seat trying find booster fit between car-seats

These keywords have been ranked in terms of their word frequency from high frequency to low frequency, such that the first keyword has the highest word frequency. We can get a global transition frequency matrix as follows:

	noise	seat	fit	baby	use
noise	10	1	3	0	0
seat	1	0	0	0	0
fit	3	0	0	0	0
baby	0	0	0	0	0
use	0	0	0	0	0

Secondly, we calculate global transition probabilities. To find the transition probabilities, the frequencies are divided by the sum of the values in that row. For example, we use different number to represent noise and different keywords. We get global transition probabilities as following:

	noise	seat	fit	baby	use
noise	$Q_{0,0}$ =0.714	$Q_{0,1}$ =0.071	$Q_{0,2}$ =0.215	$Q_{0,3}$ =0.0	$Q_{0,4}$ =0.0
seat	$Q_{1,0}$ =1.0	$Q_{1,1}$ =0	$Q_{1,2}$ =0.0	$Q_{1,3}$ =0.0	$Q_{1,4}$ =0.0
fit	$Q_{2,0}$ =1.0	$Q_{2,1}$ =0.0	$Q_{2,2}$ =0.0	$Q_{2,3}$ =0.0	$Q_{2,4}$ =0.0
baby	$Q_{3,0}$ =0.0	$Q_{3,1}$ =0.0	$Q_{3,2}$ =0.0	$Q_{3,3}$ =0.0	$Q_{3,4}$ =0.0
use	$Q_{4,0}$ =0.0	$Q_{4,1}$ =0.0	$Q_{4,2}$ =0.0	$Q_{4,3}$ =0.0	$Q_{4,4}$ =0.0

Table 17: Global Transition Probabilities

Similarly, individual transition probabilities are:

Answer 1

$P_{0,0} = 0.75$	$P_{0,1} = 0.0$	$P_{0,2} = 0.25$	$P_{0,3} = 0.0$	$P_{0,4} = 0.0$
$P_{1,0} = 0.0$	$P_{1,1} = 0.0$	$P_{1,2} = 0.0$	$P_{1,3} = 0.0$	$P_{1,4} = 0.0$
$P_{2,0} = 1.0$	$P_{2,1} = 0.0$	$P_{2,2} = 0.0$	$P_{2,3} = 0.0$	$P_{2,4} = 0.0$
$P_{3,0} = 0.0$	$P_{3,1} = 0.0$	$P_{3,2} = 0.0$	$P_{3,3} = 0.0$	$P_{3,4} = 0.0$
$P_{4,0} = 0.0$	$P_{4,1} = 0.0$	$P_{4,2} = 0.0$	$P_{4,3} = 0.0$	$P_{4,4} = 0.0$

Answer 2

$P_{0,0} = 0.667$	$P_{0,1} = 0.167$	$P_{0,2} = 0.167$	$P_{0,3} = 0.0$	$P_{0,4} = 0.0$
$P_{1,0} = 1.0$	$P_{1,1} = 0.0$	$P_{1,2} = 0.0$	$P_{1,3} = 0.0$	$P_{1,4} = 0.0$
$P_{2,0} = 1.0$	$P_{2,1} = 0.0$	$P_{2,2} = 0.0$	$P_{2,3} = 0.0$	$P_{2,4} = 0.0$
$P_{3,0} = 0.0$	$P_{3,1} = 0.0$	$P_{3,2} = 0.0$	$P_{3,3} = 0.0$	$P_{3,4} = 0.0$
$P_{4,0} = 0.0$	$P_{4,1} = 0.0$	$P_{4,2} = 0.0$	$P_{4,3} = 0.0$	$P_{4,4} = 0.0$

5.2.3 Model

When we get the **global transition probability matrix** and the **individual transition probability matrix**, we derive a new entropy methodology in terms of these probabilities. For row i, the **Transition Probability Entropy**, $M_n^i(\boldsymbol{P})$, is:

$$M_n^i(\boldsymbol{P}) = -\sum_{j=0}^{n} Q_{i,j} \times P_{i,j} \times Log\left(P_{i,j}\right), \qquad (5-1)$$

where, $i = 0,1, \dots, n$. In the following, we will use row 0 to refer to the noise words and row i to refer to keywords (i.e. assume $i > 0$ unless otherwise stated). The **Total Transition Probability Entropy** is:

$$M_n(\boldsymbol{P}) = \sum_{i=0}^{n} M_n^i(\boldsymbol{P}) . \qquad (5-2)$$

The transition probability entropy for the row i contributes to find the trend of the transition from the keyword i to other keywords. Different keywords will show different value of transition probability entropy. Thus, we can use this methodology to check which keywords' transition an individual answer shows. The Markov transition probability entropy contributes to assess answers from the global and individual transition probabilities of keywords and noise.

Since the **Total Transition Probability Entropy** is similar to the general entropy in Chapter 4, we imitate work in Chapter 4 to show some propositions:

Proposition 4: For any answers,

(1) If $0 < P_{0,0} < 1$, then $M_n(\boldsymbol{P}) > 0$

(2) $M_n(\boldsymbol{P}) = 0$ if and only if $M_n^i(\boldsymbol{P}) = 0, i = 0,1, \dots, n$. In addition, for some i, if $M_n^i(\boldsymbol{P}) = 0$, then

(2-1) $P_{i,j} = 0 \; for \; j = 0,1, \dots, n.$

or

(2-2) there exists a j_0, such that $P_{i,j_0} = 1 \; and \; P_{i,j} = 0, for \; j = 0,1, \dots, n, \; j \neq j_0$.

Proof:

(1) First of all, $M_n(\boldsymbol{P}) = \sum_{i=0}^{n} M_n^i(\boldsymbol{P}) = -Q_{0,0} \times P_{0,0} \times Log(P_{0,0}) - Q_{0,j} \times P_{0,j} \times Log(P_{0,j}) - Q_{i,0} \times P_{i,0} \times Log(P_{i,0}) - \sum_{i,j=1}^{n} Q_{i,j} \times P_{i,j} \times Log(P_{i,j})$.

Secondly, since $Q_{0,0} > 0$ and $0 < P_{0,0} < 1$, then $-Q_{0,0} \times P_{0,0} \times Log(P_{0,0}) > 0$. On the other hand, other items in this formula is larger than or equal to 0. Thus, we get $M_n(\boldsymbol{P}) > 0$.

(2) First of all, we know $M_n(\boldsymbol{P}) = \sum_{i=0}^{n} M_n^i(\boldsymbol{P}) = -\sum_{j=0}^{n} Q_{i,j} \times P_{i,j} \times Log(P_{i,j}) \geq 0$. If $M_n(\boldsymbol{P}) = 0$, then $\sum_{i=0}^{n} M_n^i(\boldsymbol{P}) = 0$. Thus, $M_n^i(\boldsymbol{P}) = 0, for \; i = 0,1, \dots, n$. Secondly, if $M_n^i(\boldsymbol{P}) = 0, i = 0,1, \dots, n$, then $M_n(\boldsymbol{P}) = \sum_{i=0}^{n} M_n^i(\boldsymbol{P}) = 0$.

In addition, if $M_n^i(\boldsymbol{P}) = -\sum_{j=0}^{n} Q_{i,j} \times P_{i,j} \times Log(P_{i,j}) = 0$, we get $-Q_{i,j} \times P_{i,j} \times Log(P_{i,j}) = 0$. Thus, there are two following situations:

(a) $Q_{i,j} = 0\ for\ j = 0,1, \dots, n$. It means $P_{i,j}$ is equal to 0 for $j = 0,1, \dots, n$.

(b) When $Q_{i,j} \neq 0$, we can make the following analysis:

$M_n^i(\boldsymbol{P}) = -\sum_{j=0}^{n} Q_{i,j} \times P_{i,j} \times Log(P_{i,j}) = -Q_{i,0} \times P_{i,0} \times Log(P_{i,0}) - Q_{i,1} \times P_{i,1} \times Log(P_{i,1}), \dots, -Q_{i,j_0-1} \times P_{i,j_0-1} \times Log(P_{i,j_0-1}) - Q_{i,j_0+1} \times P_{i,j_0+1} \times Log(P_{i,j_0+1}), \dots, -Q_{i,n} \times P_{i,n} \times Log(P_{i,n}) - Q_{i,j_0} \times P_{i,j_0} \times Log(P_{i,j_0})$. If $P_{i,j_0} = 1$, we get $-Q_{i,j_0} \times P_{i,j_0} \times Log(P_{i,j_0}) = 0$. If $P_{i,j} = 0, i, j = 0,1, \dots, n, j \neq j_0$, we have $-Q_{i,j} \times P_{i,j} \times Log(P_{i,j}) = 0$. Thus, we get $M_n^i(\boldsymbol{P}) = -\sum_{j=0}^{n} Q_{i,j} \times P_{i,j} \times Log(P_{i,j}) = 0$.

Sometimes, though an answer contains many keywords, $M_n^i(\boldsymbol{P})$ of this answer is still equal to zero. Here, we give some examples to show $M_n^i(\boldsymbol{P}) = 0$ for the row i in following remarks:

1) There is only one noise, other words are keywords, and one keyword does not appear twice. For example, the digitalized answer vector is “1230456”.
2) There are no noise words and one keyword does not appear twice. For example, the digitalized answer vector is “1234”.
3) There are no noises and the beginning and end locations are the same keyword. If there are other keywords, they do not appear twice. For example, the digitalized answer vector is “1231”.
4) The number of words in an answer is three and there is at least one word different to other two words. If there are two words at the beginning, these two words are not the same keyword. For example, the digitalized answer vector is “102, 011, 202”.

Additionally, for a keyword i, if $P_{i,j} = 0$, the probability of this keyword may not be zero. For example, if there is only one word in an answer and this word is a keyword, the transition

probability from this keyword to other words is zero. But, the individual probability of this keyword in the answer is 1. The global probability of this keyword is not zero.

Analogously, when we obtain global transition probabilities from one word to another word, we can imitate the **Definition 2** and the **Theorem 1** in Chapter 4 to present following definition and theorem.

Definition 7: Given global transition probabilities: $\{Q_{0,0}, Q_{0,1}, \dots Q_{0,n}, Q_{1,0}, \dots, Q_{n,n}\}$, the maximum transition entropy probabilities matrix $B_{nn} \coloneqq [\overrightarrow{B_0}, \overrightarrow{B_1}, \dots, \overrightarrow{B_n}]$, where $\overrightarrow{B_i} \coloneqq [B_{i,0}, B_{i,1}, \dots, B_{i,n}]^T$ with $\sum_{k=0}^{n} B_{ik} = 1$, $\overrightarrow{B_i}$ is defined by

$$\overrightarrow{B_i} = \underset{\boldsymbol{P_i}}{\operatorname{argmax}} E_{in}(\boldsymbol{P_i}),$$

where, $\boldsymbol{P_i} \coloneqq [P_{i,0}, P_{i,1}, \dots, P_{i,n}]^T$ with $\sum_{k=0}^{n} P_{ik} = 1$, $E_{in}(\boldsymbol{P_i}) = -\sum_{j=0}^{n} Q_{i,j} \times P_{i,j} \times Log(P_{i,j})$. Then, we have the following theorem for $\overrightarrow{B_i}$.

Theorem 2: Suppose the number of total different transitions $n + 1 \geq 3$, and for each row i, $\{Q_{i,0}, Q_{i,1}, \dots, Q_{i,n}\}$ contains at least three non-zero elements. Then, there exist the maximum transition entropy probabilities $\overrightarrow{B_i} \coloneqq [B_{i,0}, B_{i,1}, \dots, B_{i,n}]^T$, so that $\overrightarrow{B_i} = \underset{\boldsymbol{P_i}}{\operatorname{argmax}} E_{in}(\boldsymbol{P_i})$ and $B_{i,j} = e^{-1-\frac{\lambda_j^i}{Q_{i,j}}}, j = 0,1, \dots, n$, where $\lambda_j^i > 0$ is a unique positive value and $\sum_{j=0}^{n} B_{i,j} = 1$ and $B_{i,j} = 0$ if $Q_{i,j} = 0$.

Proof: When $\{Q_{i,0}, Q_{i,1}, \dots, Q_{i,n}\}$ are given, in order to maximize $E_{in}(\boldsymbol{P_i})$, we define a function as following:

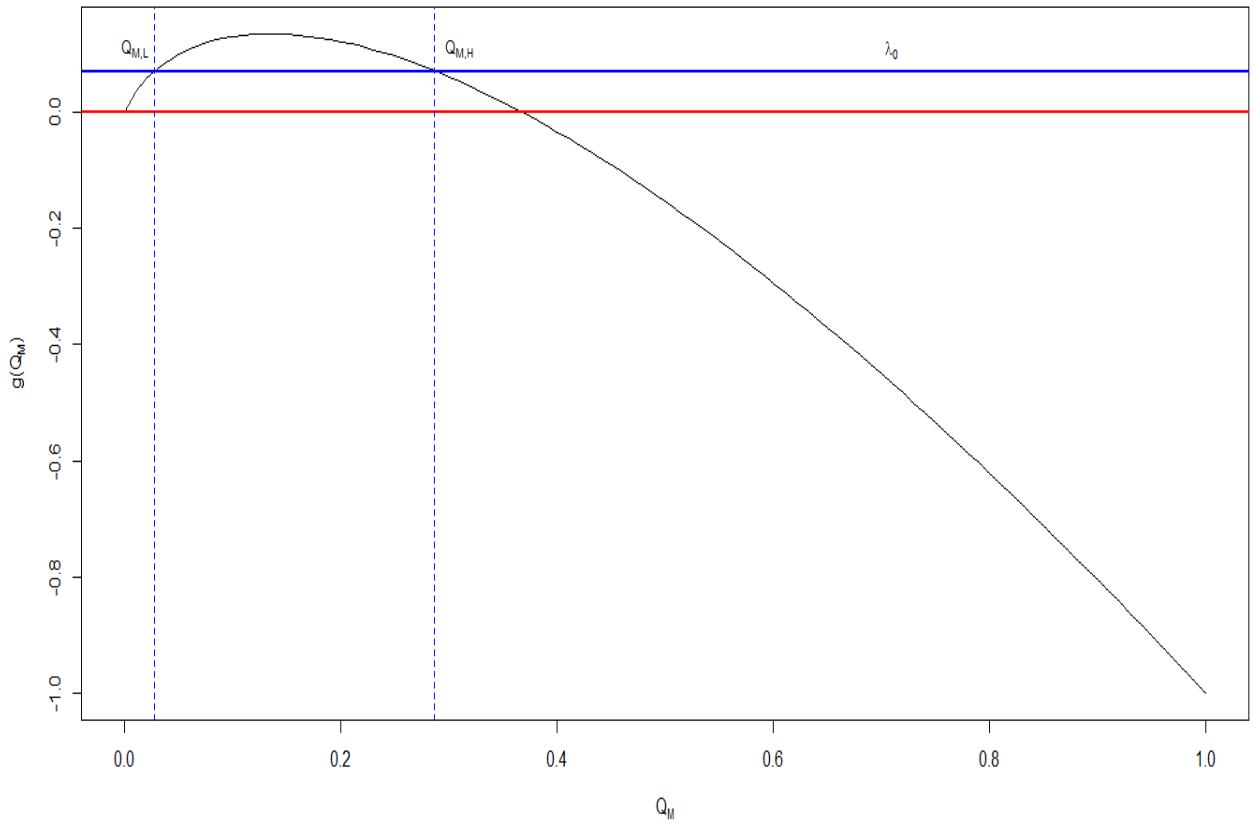
$$f(P_{i,0}, P_{i,1}, \dots, P_{i,n}, \lambda_j^i) = -\sum_{i,j=0}^{n} Q_{i,j} \times P_{i,j} \times Log(P_{i,j}) + \lambda_j^i(1 - P_{i,0} - P_{i,1} -, \dots, -P_{i,n}).$$

Without loss of generality, we assume $Q_{i,j} > 0$ for j from 0 to n. If we want to make $\frac{\partial f}{\partial P_{i,j}} = -\big(\log(P_{i,j}) + 1\big)Q_{i,j} - \lambda_j^i = 0$, we can get $\widehat{P_{i,J}} = e^{-1-\frac{\lambda_j^i}{Q_{i,j}}}$. We define $B_{i,j} \coloneqq \widehat{P_{i,J}}$, then $B_{i,j} = e^{-1-\frac{\lambda_j^i}{Q_{i,j}}}$. Thus, we can use $B_{i,j}, j = 0,1, \dots, n.$ to make $E_{in}(\boldsymbol{P_i})$ to be maximum. Since $\sum_{j=0}^{n} \widehat{P_{i,J}} = 1$, we get $\sum_{j=0}^{n} e^{-1-\frac{\lambda_j^i}{Q_{i,j}}} = 1$. We can define a function $g\big(\lambda_j^i\big) = \sum_{j=0}^{n} e^{-1-\frac{\lambda_j^i}{Q_{i,j}}} - 1$, then, we get $g'\big(\lambda_j^i\big) = \sum_{j=0}^{n} -\frac{1}{Q_{i,j}} e^{-1-\frac{\lambda_j^i}{Q_{i,j}}} < 0, for\ \lambda_j^i \geq 0$. It means $g\big(\lambda_j^i\big)$ is monotone decreasing for $\lambda_j^i \geq 0$. Since $g(\infty) = -1$ and $g(0) = \frac{n+1}{e} - 1 > 0, for\ n \geq 2$. Therefore, we can find a unique positive λ_0^i to make $g\big(\lambda_0^i\big) = 0$. It means we can get a unique $B_{i,j} = e^{-1-\frac{\lambda_0^i}{Q_{i,j}}}, j = 0,1, \dots, n$. That is a vector $\overrightarrow{B_i} \coloneqq [B_{i,0}, B_{i,1}, \dots, B_{i,n}]^T$. Therefore, for $i = 0,1, \dots, n.$ we can define the matrix of the maximum transition entropy probabilities $B_{nn} \coloneqq [\overrightarrow{B_0}, \overrightarrow{B_1}, \dots, \overrightarrow{B_n}]$.

When we get $B_{i,j}$ and $Q_{i,j}$, we can also imitate (4-2) to analyze them. For the row i, we have $B_{i,j} = e^{-1-\frac{\lambda_0^i}{Q_{i,j}}}, for\ 0 \leq j \leq n$ and $\lambda_0^i > 0$, we try to compare $B_{i,j}$ and $Q_{i,j}$ in terms of λ_0^i. For the row i, we firstly design a function as following:

$$h(Q_M) = (-logQ_M - 1)Q_M, \qquad (5-3)$$

where, $0 < Q_M < 1$. We plot (5-3) as following:



The red line represents $h(Q_M) = 0$. If we choose a $0 < \lambda_0^i < \max_{0<Q_M<1} h(Q_M)$, we can draw a blue line in the figure to represent it. There are still two points of intersection, the first one point is $Q_{M,L}$; the second one point is $Q_{M,H}$. Here, we can also compare $Q_{i,j}$ and $B_{i,j}$ in following remarks:

For the row i,

(1) if $Q_{M,L} < Q_{i,j} < Q_{M,H}$, then $\lambda_0 < h(Q_{i,j}) = (-logQ_{i,j} - 1)Q_{i,j}$. We can also deduce this inequation in following way:

$$\lambda_0 < (-logQ_{i,j} - 1)Q_{i,j} \Rightarrow \frac{\lambda_0}{Q_{i,j}} < (-logQ_{i,j} - 1)$$

$$\Rightarrow logQ_{i,j} < -1 - \frac{\lambda_0}{Q_{i,j}} \Rightarrow Q_{i,j} < e^{-1-\frac{\lambda_0}{Q_{i,j}}} = B_{i,j}, \qquad (5-4)$$

(2) if $0 < Q_{i,j} < Q_{M,L}$ or $Q_{M,H} < Q_{i,j}$, then $h(Q_{i,j}) = (-logQ_{i,j} - 1)Q_{i,j} < \lambda_0$. We can also deduce this inequation in following way:

$$(-logQ_{i,j} - 1)Q_{i,j} < \lambda_0 \Rightarrow (-logQ_{i,j} - 1) < \frac{\lambda_0}{Q_{i,j}}$$

$$\Rightarrow -1 - \frac{\lambda_0}{Q_{i,j}} < logQ_{i,j} \Rightarrow B_{i,j} = e^{-1-\frac{\lambda_0}{Q_{i,j}}} < Q_{i,j}. \quad (5-5)$$

Similarly, we hope the transition probability is as high as possible. But, if $Q_{M,H} < Q_{i,j}$, the transition is demoted. This situation may be reasonable. Because, if a transition is repeated too many times, it will mislead readers to focus on this transition and ignore other transitions. It means that the transition probability should not be too large. Actually, if the transition between two keywords happen too many times, we can regard these two keywords as an integral whole. It means these two keywords represent one keyword. Thus, to some extent, we can avoid the transition to be demoted. In order to understand the meaning, an example will be given in the following contents.

Here, we imitate the definition in Chapter 4 to give a definition about the demotion or promotion of a transition, this definition can be used to describe the transition of words:

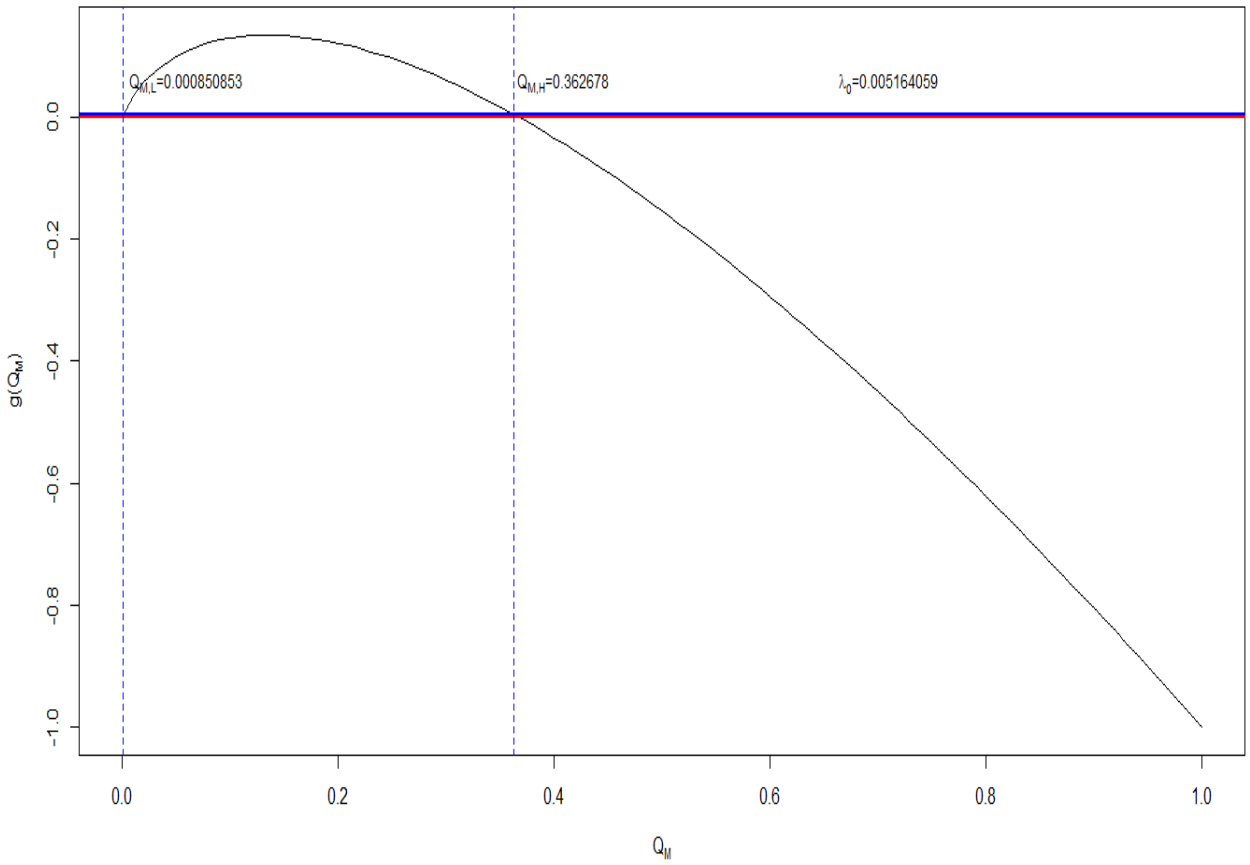
Definition 8: Given $Q_{i,j}$ and $B_{i,j}, 0 \leq i \leq n$ and $0 \leq j \leq n$. If $B_{i,j} < Q_{i,j}$, then the transition from the keyword i to the keyword j is called "demotion" for the transition. If $B_{i,j} > Q_{i,j}$, then the transition from the keyword i to the keyword j is called "promotion" for the transition.

5.3 Amazon case study

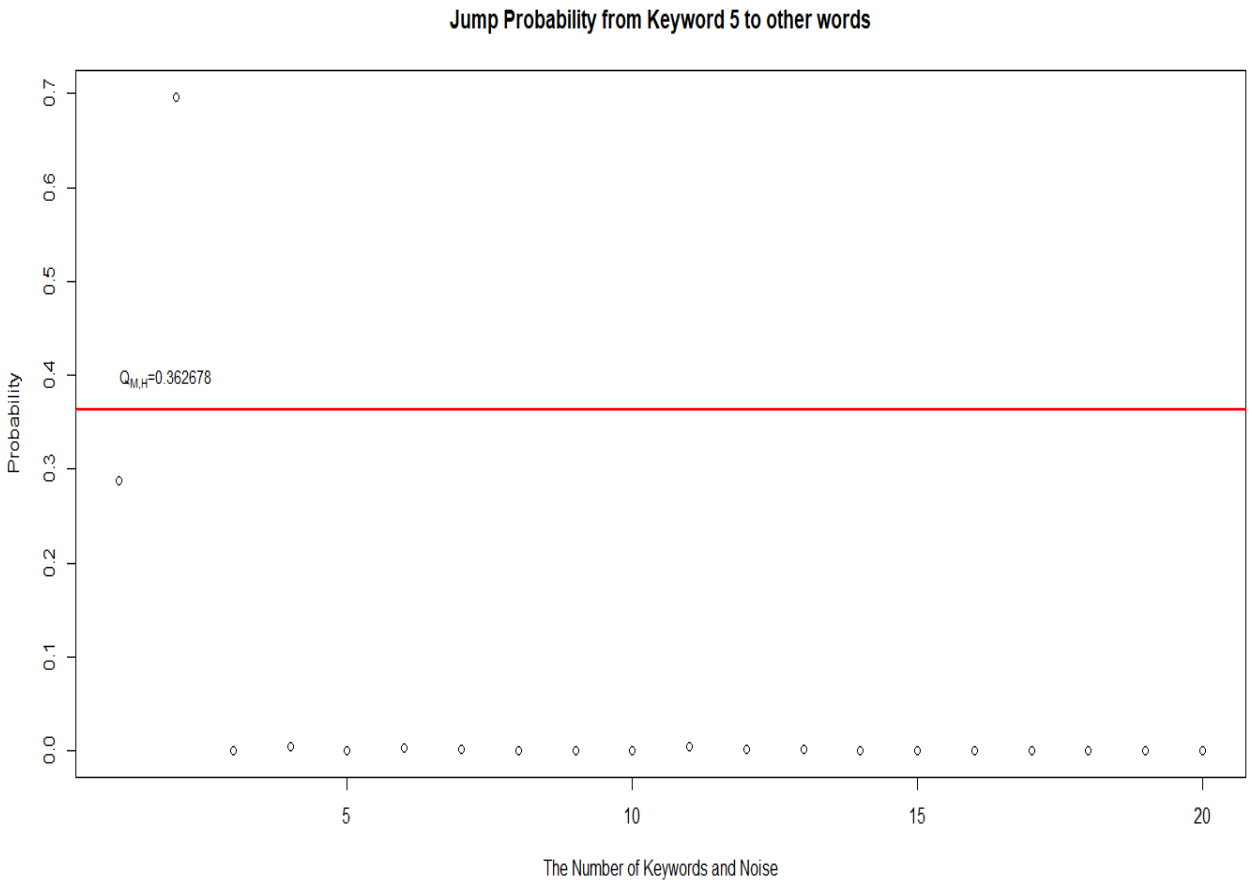
We still use Amazon answers (see Chapter 6 for a description of the data) to analyze $Q_{i,j}$. We adapt 19 keywords, which are determined in Chapter 4 to analyze answers. For a keyword $i, 1 \leq i \leq 19$, most $Q_{0,i}$ and $Q_{i,0}$ are large. Most $Q_{i,j}\ for\ i,j = 1,2,\ldots,19$ are very small. However, there is a special transition probability from the keyword 5 ("car") to the keyword 1 ("seat"), $Q_{5,1}$, which is distinct to other transition probabilities. In this section, we demonstrate our method by analyzing this special case.

5.3.1 How to judge two keywords as a pair?

We firstly use (5-3) to calculate the value of $Q_{M,L}$ and $Q_{M,H}$ respectively. Secondly, we plot (5-3), $Q_{M,L}$, and $Q_{M,H}$ as following:



In above figure, $Q_{M,L} = 0.000850853$ and $Q_{M,H} = 0.362678$. If we compare $Q_{5,j}\ for\ j = 0,1,2,\ldots,19$ and $Q_{M,H}$, we can get relationships as following:

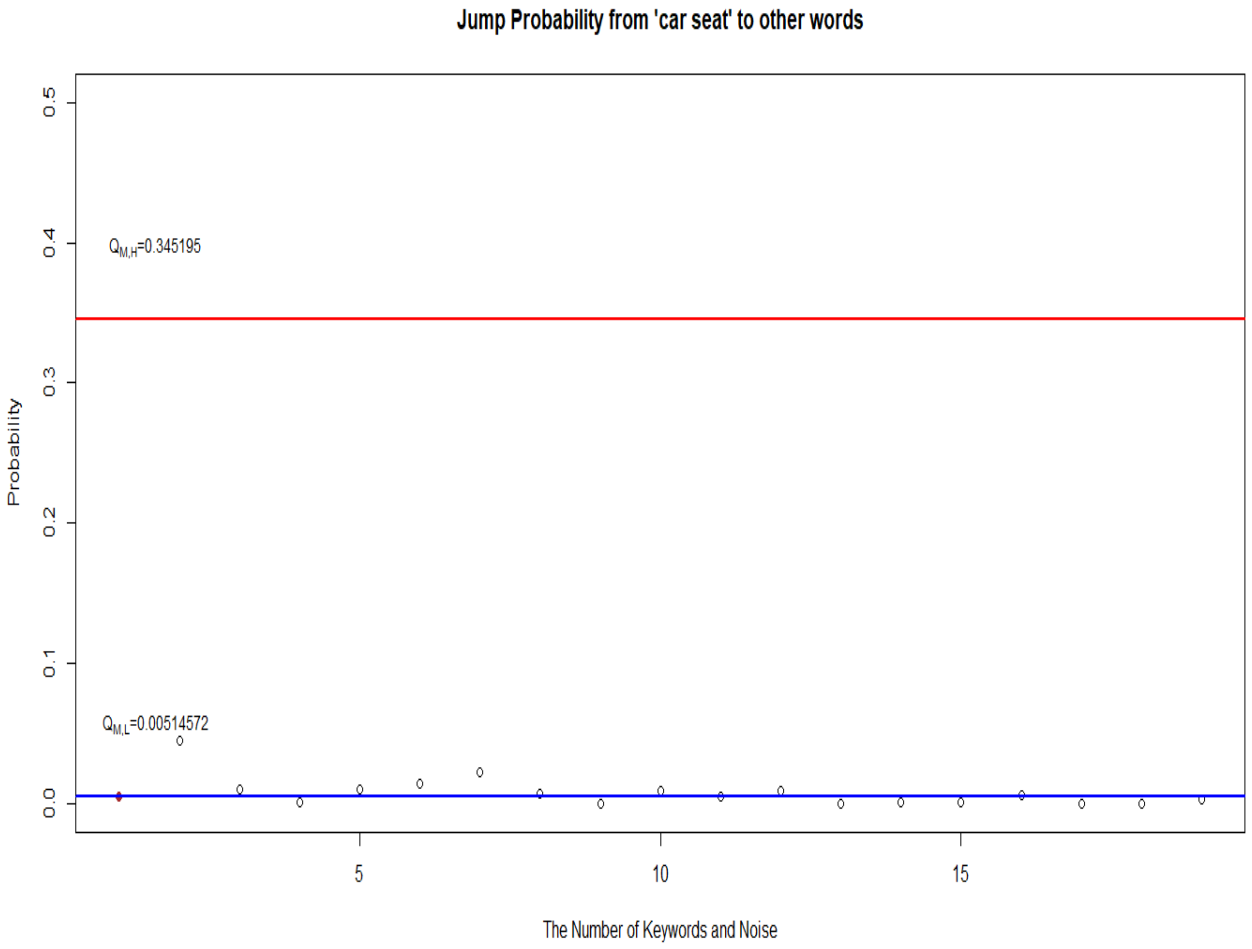


From above figure, we find that $Q_{5,1} > Q_{M,H}$, then $B_{5,1} < Q_{5,1}$ in terms of (5-5). It means the transition from the keyword 5 to 1 is a "demotion". Correspondingly, the value of entropy is reduced. We want to investigate why the value of entropy is reduced. Now, we calculate the number of answers, which contain "car" and "car seat" respectively:

Total Number of Answers with "car"	Total Number of Answers with "car seat"
1193	658

In all answers with the keyword 5 ("car"), there are 55.16% answers which actually have the pair of "car seat". This percentage gives us a likelihood that we can regard these two keywords as a pair in the process of our analysis. Therefore, we try to insert "seat" after "car" if there is no "seat" after this "car" originally. Then, we re-digitalize answers by regarding "car" or "car seat" as the same keyword. Here, the keyword "seat" will not be replaced. Also, we do not change other

original keywords and their location in original rank sequence. But, when “car seat” appears, we only calculate the transition probability from “car seat” to other words. We calculate $Q_{M,L}, Q_{M,H}$ and plot global transition probabilities from the keyword “car seat” to other words:



Compared with the original $Q_{5,1}$, we find that the global transition probability from the keyword 5 (“car seat”) to the keyword 1 (“seat”) is less than $Q_{M,H}$.

5.3.2 Compare the global transition probability and the maximum transition entropy probability

Now, we have used the new keyword (“car seat”) to replace the original keyword “car”. We analyze relationships between $Q_{i,j}$ and $B_{i,j}$:

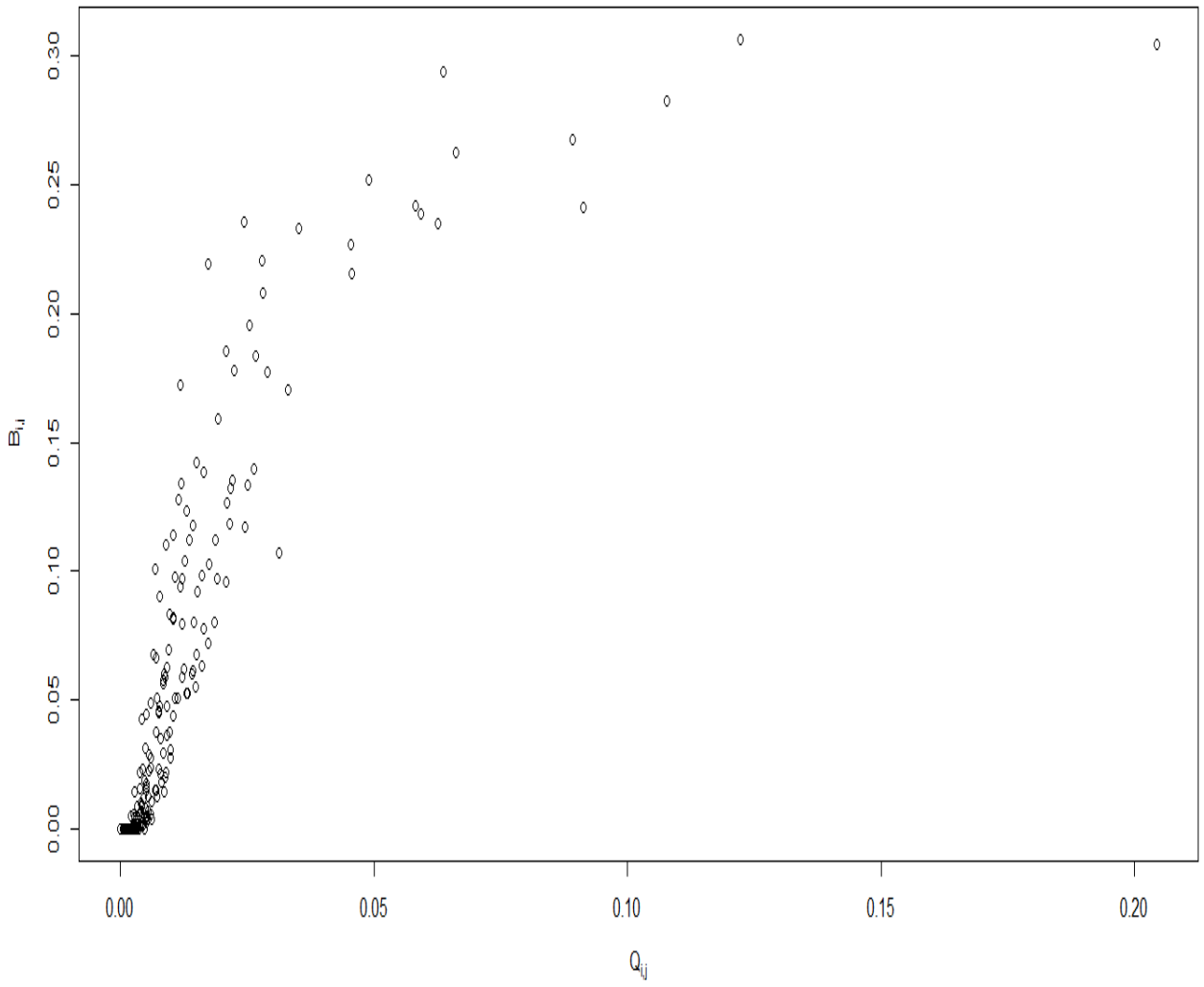


Figure 33: Relationship between global transition probability and maximum entropy transition probability

Figure 33 shows relationships between $Q_{i,j}$ and $B_{i,j}$ for all 19 keywords. These relationships accord with the rules which are described in (5-4) and (5-5). We can see that higher transition probabilities, Q_{ij}, induce higher maximum transition entropy probabilities, $B_{i,j}$.

5.3.3 Applying the transition probability to a small number of comments for a real Amazon product

We want to use a real Amazon product to check our methodology in order to demonstrate our methodology in practice. The global transition probability can help us to judge the importance of

a transition from one keyword to another keyword. Here, we still adapt the same Amazon product in Chapter 4 (See 4.3.4). We also use the same keywords, which are selected by the general entropy for this product. Some Amazon keywords of this product contain more than or equal to two keywords. When we remove stopping words, these Amazon keywords can be considered as a transition one keyword to another keyword. We can call these Amazon keywords as **Pair-Keywords**.

Our previous analysis of Amazon answers is based on the large amount of data. But, the number of comments for this product is small. Since the number of keywords, which are selected by the general entropy, is 12, we can calculate their transition global probabilities:

	noise	keyword 1: quality	keyword 2: good	…	keyword 12: them
noise	0.827	0.011	0.023	…	0.011
keyword 1: quality	0.645	0.0	0.065	…	0.0
keyword 2: good	0.516	0.452	0.0	…	0.0
……					
keyword 12: them	0.889	0.0	0.0	…	0.0

Table 18: Global Transition Probabilities of an Amazon product

According to Table 18, when we obtain $Q_{i,j}, 0 \leq i \leq 12$, we can get λ_i for each row:

$\lambda_0 = 0.028$	$\lambda_7 = 0.021$
$\lambda_1 = 0.057$	$\lambda_8 = 0.03$

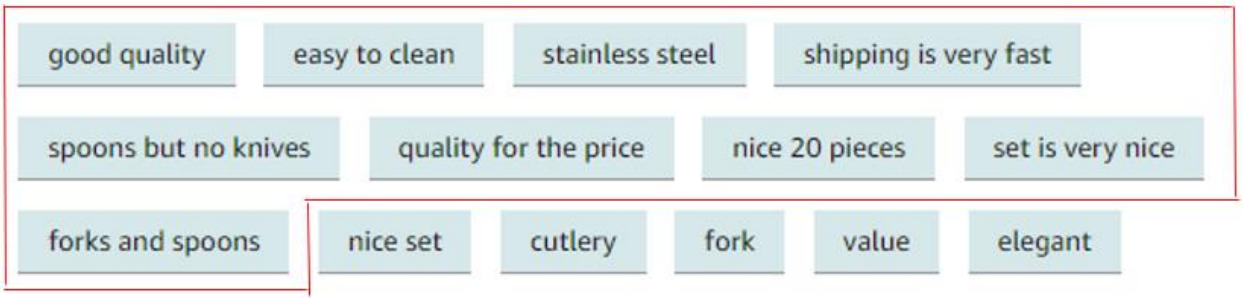
$\lambda_2 = 0.009$	$\lambda_9 = 0.039$
$\lambda_3 = 0.006$	$\lambda_{10} = 0.071$
$\lambda_4 = 0.009$	$\lambda_{11} = 0.012$
$\lambda_5 = 0.055$	$\lambda_{12} = 7.451e-05$
$\lambda_6 = 0.073$	

Then, we can get $B_{i,j}$ in terms of λ_i and $Q_{i,j}$ as following:

$B_{0,0} = 0.356$	$B_{0,1} = 0.03$	…	$B_{0,12} = 0.03$
$B_{1,0} = 0.337$	$B_{1,1} = 0.0$	…	$B_{1,12} = 0.0$
……			
$B_{12,0} = 0.368$	$B_{12,1} = 0.0$	…	$B_{12,12} = 0.0$

From above table, we find that $\sum_{j=0}^{12} B_{i,j} = 1$ for $0 \leq i \leq 11$. But, $\sum_{j=0}^{12} B_{12,j} < 1$. The reason is that, the keyword "them" only transfers to noises and to the keyword "not" in comments. Based on our constraints, the keyword "them" should not be used to analyze transition probabilities. This also illustrates that the data volume should be as large as possible so that each keyword can make a variety of transitions.

Since some Amazon keywords are Pair-Keywords, we can compare transitions of Pair-Keywords and transitions of keywords selected by the general entropy. Here, we only analyze transitions between two keywords.



For the row i, when λ_i is obtained, we can obtain $Q_{M,L}$ and $Q_{M,H}$ by the formula (5-3). Because if $Q_{M,L} < Q_{i,j} < Q_{M,H}$, then $Q_{i,j} < B_{i,j}$ for the keyword i and the keyword j. We decide to select two keywords, the transition probability of which is between $Q_{M,L}$ and $Q_{M,H}$ (**Note:** we do not analyze transitions between one keyword and noise). The following table demonstrates how the keywords selected by the general entropy method agree with the pair-keywords suggested by Amazon:

Keyword 1	**Keyword 2**	**The transition probability from keyword 1 to keyword 2**	$Q_{M,L}$	$Q_{M,H}$	**Pair-Keywords**
quality	price	0.161	0.019	0.305	quality for the price
set	nice	0.12	0.002	0.359	set is very nice
forks	spoons	0.133	0.005	0.346	forks and spoons
nice	set	0.167	0.011	0.326	nice set
spoons	but	0.083	0.002	0.356	spoons but no knives

Transitions of these keywords can be seen to match Pair-Keywords. Though there are some other transition probabilities of keywords between $Q_{M,L}$ and $Q_{M,H}$, no Pair-Keywords are matched with these keywords. We also analyze some other Amazon products and obtain similar outcomes. Here,

a special case is that the transition probability from the keyword "good" to the keyword "quality" is equal to 0.45161290322580644, which is larger than $Q_{M,H}$. These two keywords can be merged together to be one keyword. Based on above analysis, the transition probability of two keywords can help us to find which keywords are frequently transferred from one to another. The last row of the table is an interesting case. The pair-keywords selected by Amazon are "spoons but no knives," but our method picked "spoons" and "but." Our method of selecting keywords is performed algorithmically, which means that it does not always choose what would be logical to humans.

5.4 Conclusion

Different from the analysis in Chapter 4, this chapter mainly focuses on transition probabilities. We extend methodologies in Chapter 4 to analyze answers with inner connection of keywords in mind. We first introduce the Total Transition Probability Entropy and analyze its propositions. Also, we propose a definition and theorem. For each keyword, we calculate $Q_{M,L}$ and $Q_{M,H}$. Then, we obtain similar conclusion as in Chapter 4. When $Q_{M,L} < Q_{i,j} < Q_{M,H}$, the maximum transition entropy, $B_{i,j}$, will be larger than $Q_{i,j}$. Otherwise, the maximum transition entropy, $B_{i,j}$, will be smaller than $Q_{i,j}$.

The main contribution of the maximum transition entropy is that we can use it to judge which information transition is important in speech. If the speech is found to transfer from one keyword to another keyword with low repetition rates in an answer, we can believe that this information transition is not important. However, if the speech is found to transfer from one keyword to another keyword with high repetition rates in an answer, we can regard these two keywords as one keyword. Our methodology provides two thresholds: $Q_{M,L}$ and $Q_{M,H}$ to check the importance of transition. For keywords i and j, if $Q_{M,L} < Q_{i,j} < Q_{M,H}$, we can believe the transition from the keyword i to the keyword j is important. There is an important constraint for our methodology: if we want to calculate the maximum transition entropy, one word should be at least transferred to three different keywords or noise.

The methodology of the Total Transition Probability Entropy can be applied in many ways. Since different people have different experiences, the methodology can be used to distinguish different speakers according to their trends of speech. Also, it may help human resources specialists to find proper interviewees in the job interview. In the future, we can analyze transitions between more than two words. Thus, we can analyze propositions of speeches or documents more accurately.

Chapter 6

6 Methodologies Comparison

In this chapter, we compare several developed methodologies to evaluate their performances and differences. We also analyze the length distribution of answers to find the relationship between answer length and developed methodologies. Random pattern is another feature of answers. It can reflect the distribution of answers. Thus, we also compare Wald–Wolfowitz runs test with developed methodologies.

6.1 Data Preparation

6.1.1 Data Cleaning

We adapt following steps to clean data.

Step 1. **Obtain Answers:** we obtain original answers of "Baby" category in Amazon dataset. We save it as text files.

Step 2. **Remove Unnecessary Words**

We only use answers of Amazon data to be our analysis contents. However, these answers contain punctuation marks and some unnecessary words, which may affect analysis results. For example, some unnecessary words increase the quantity of noises and decrease percentages of keywords. We call these punctuation marks or unnecessary words as **stopping words**. In order to reduce the impact of stopping words, we remove them as many as possible. In this chapter, we regard following words as stopping words and remove them:

1. All punctuation marks
2. Articles, Prepositions, Conjunctions
3. Special symbols and numbers
4. Other unnecessary words

Punctuation marks, Articles, Prepositions, Conjunctions do not take effect the meaning of a sentence. If we remove them, we can still understand the meaning of a sentence. Thus, we can remove them. In this chapter, special symbols and numbers also do not impact our analysis, we

can remove them as well. When we find other unnecessary words, such as misspelling words, we can also remove them.

Step 1. Remove Collinearity of Words

Collinearity of words is that two different words may express the same meaning. In this thesis, if two words represent the same meaning, we can use one of them to replace another word. For example, if we choose answers of "baby" category for analysis, we use https://www.thesaurus.com/ to find all collinearities as following:

{"diminutive", "dwarf", "little", "midget", "mini", "minute", "petite", "small", "wee", "babyish", "tiny", "youthful"}

Then, we use the word "baby" to replace all synonyms.

Step 2. Remove Empty Answers

When we finish Step 1, Step 2, Step 3, and Step 4, we check all answers again and remove empty answers. We combine R and Java together to write program codes. We mainly use Java to complete following work:

- Obtain answers.
- Remove stopping words in answers.
- Remove collinearity in answers.
- Calculate CEW-DTW, KL-CEW-DTW, the General Entropy, the Transition Probability Entropy.
- Any other computations, which are easily completed by Java.

We mainly use R to complete following work:

- Analyze statistical properties.
- Calculate Dynamic Time Warping and Kullback-Leibler divergence
- Compare two methodologies about their statistical features.
- Any other computations, which are easily completed by R.

When we get answers, we calculate word frequency of all words. Then, we rank these words from high word frequency to low word frequency. We choose some words with top word frequency as keywords (e.g. top-n).

6.1.2 Obtain answers with at least two words

Since $E_n(\boldsymbol{P})$ and $M_n(\boldsymbol{P})$ of one-word answer is always 0, the total number of one-word answer is 587, which only occupy 2.7% in original answers, we can remove these one-word answers. Therefore, we decide to analyze answers, which have at least two words. The number of these answers is 21405. We call these 21405 answers to be **At-Least-Two-Words** answers. The distribution of length density of those answers is:

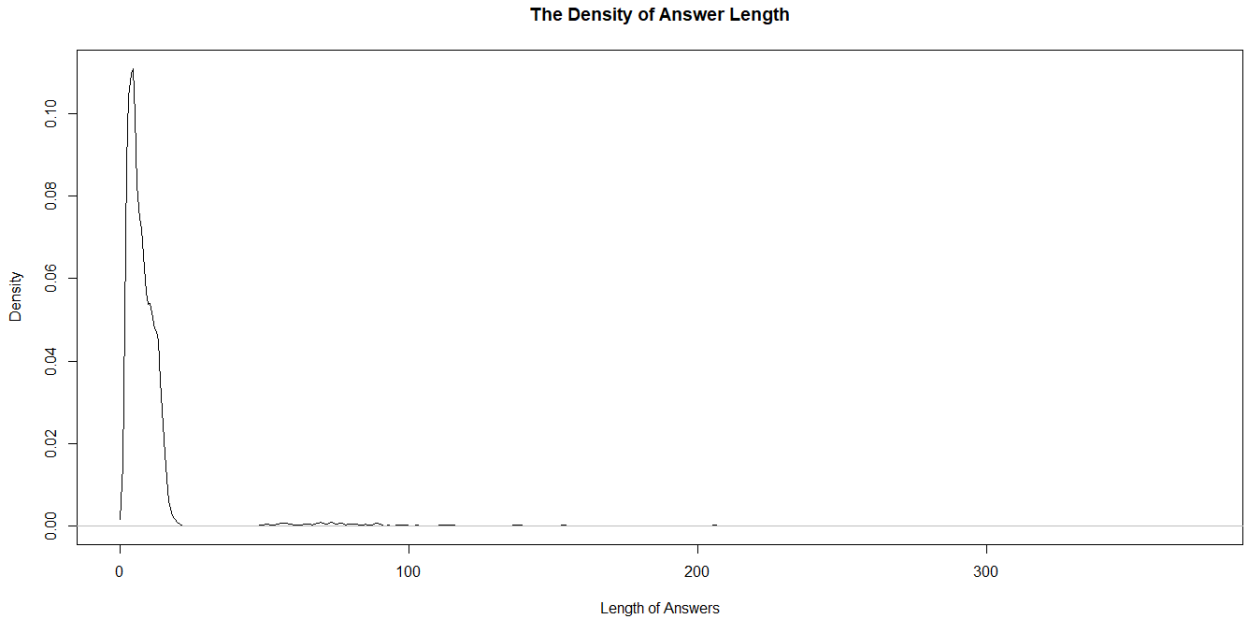


Figure 34: Length Distribution of answers

The boxplot of answers' length is:

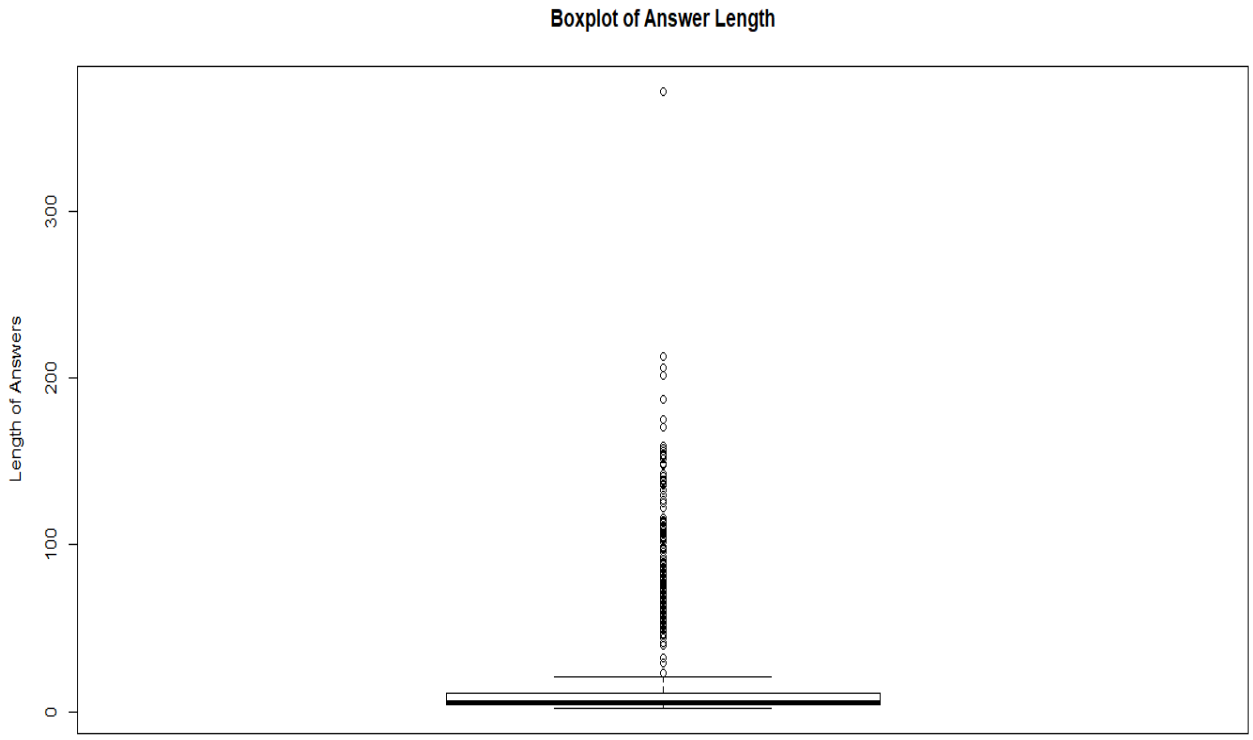


Figure 35: Boxplot of At-Least-Two-Words answers

We find that length of most answers is less than 50 words. The summary of length of answers are:

Minimum Length	1st Quantile	Median Length	Average Length	3rd Quantile	Maximum Length
2	4	6	9.203	11	372

Table 19: Summary of At-Least-Two-Words answers

6.2 Comparison of the General Entropy, the Markov Transition Probability Entropy, and CEW-DTW for each other

After introduction of data preparation, we begin to compare the General Entropy, the Markov Transition Probability Entropy, and CEW-DTW for each other by applying At-Least-Two-Words answers. We do not compare KL-CEW-DTW with other methodologies since KL-CEW-DTW require an answer to contain at least one keyword. Though these three developed methodologies

analyze answers in different ways, their computation processes are all complicated. Therefore, if we can find some relationships between them, we may use one methodology to replace others to some extent. In order to compare the relationship of these two methodologies, we adapt 19 keywords, which are used in Chapter 4.

6.2.1 Relationship between CEW-DTW and the General Entropy

6.2.1.1 Comparison of all answers

The relationship between CEW-DTW and $E_n(\boldsymbol{P})$ is:

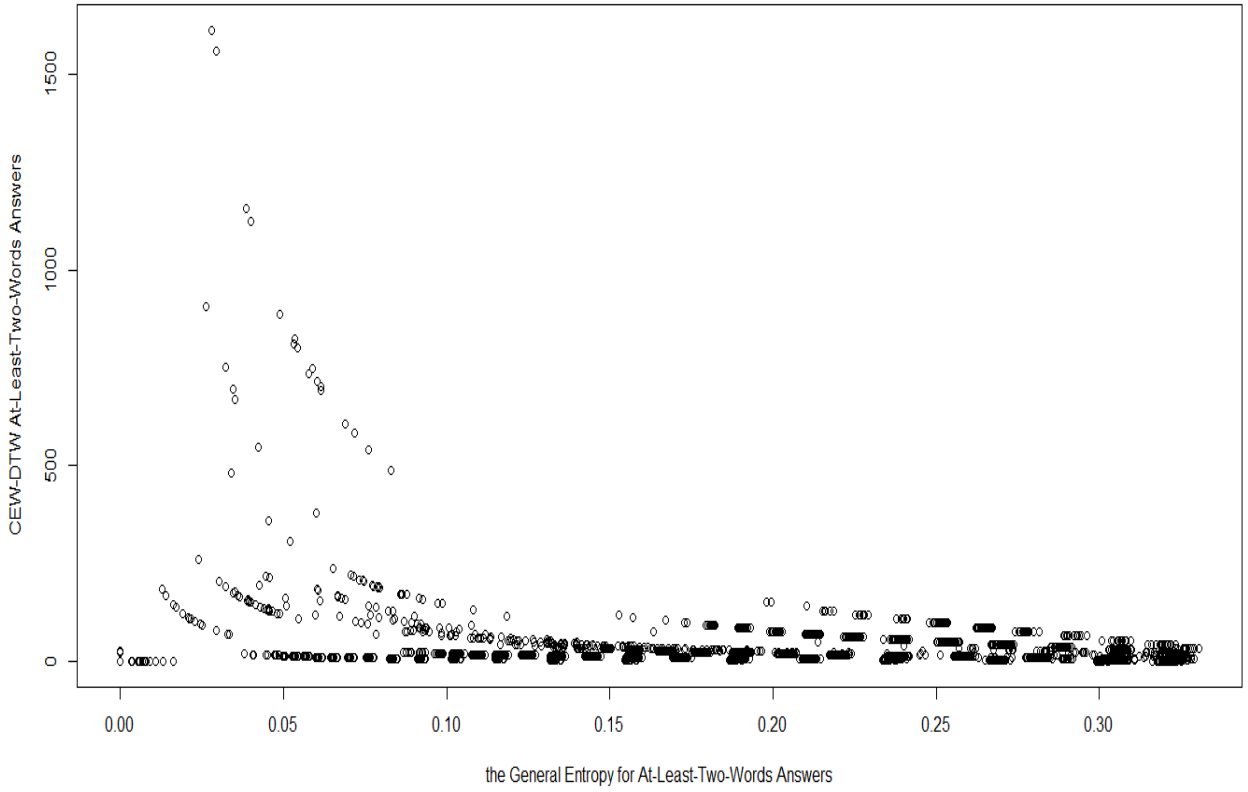


Figure 36: Relationship between CEW-DTW and the General Entropy

Figure 36 illustrates a roughly negative relationship between CEW-DTW and $E_n(\boldsymbol{P})$, but the structure of the relationship is not clear. For CEW-DTW, the high value of CEW-DTW for an answer illustrates the lower quality of this answer. But, for the general entropy, the high value of $E_n(\boldsymbol{P})$ for an answer shows the higher quality of this answer. Therefore, CEW-DTW and $E_n(\boldsymbol{P})$

can be roughly replaced for each other to verify answers' qualities to some extent. Sometimes, one of these methodologies can be instead of another.

6.2.1.2 Comparison of Answers: the percentage of noise in these answers is less than the percentage of the global noise

If an answer with $P_0 < Q_0$, it means the percentage of noise in this answer is less than the percentage of the global noise. Thus, these answers contain more information. We try to analyze the relationship between CEW-DTW and $E_n(\boldsymbol{P})$ of these answers:

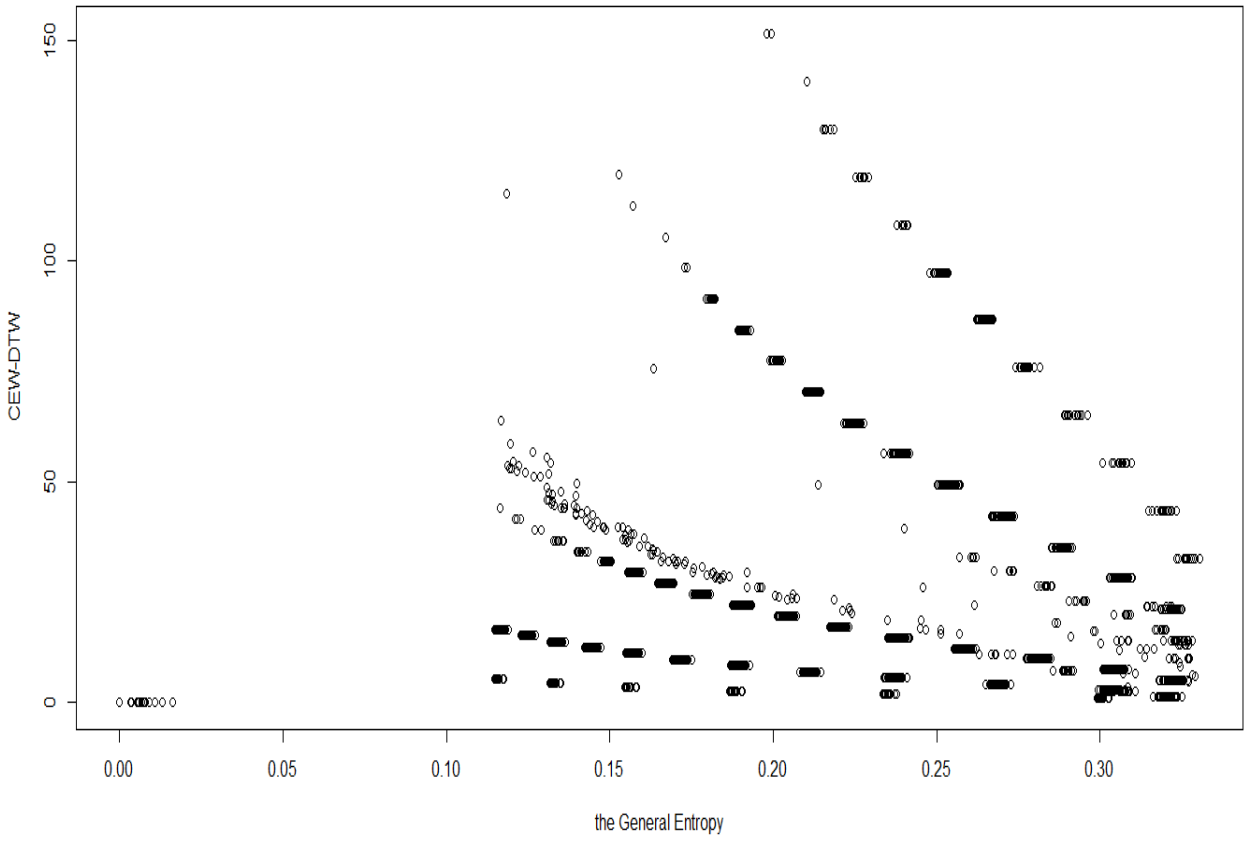


Figure 37: Relationship between CEW-DTW and the General Entropy

According to Figure 37, though these points are evidently divided into several groups, CEW-DTW and $E_n(\boldsymbol{P})$ of these answers still follow negative correlation. Therefore, they can be reciprocally

replaced for each other to test answer qualities in some way. But, we cannot strictly say they can be represented for each other to test answer qualities.

6.2.2 Relationship between CEW-DTW and the Markov Transition Probability Entropy

The relationship between CEW-DTW and $M_n(\boldsymbol{P})$ is:

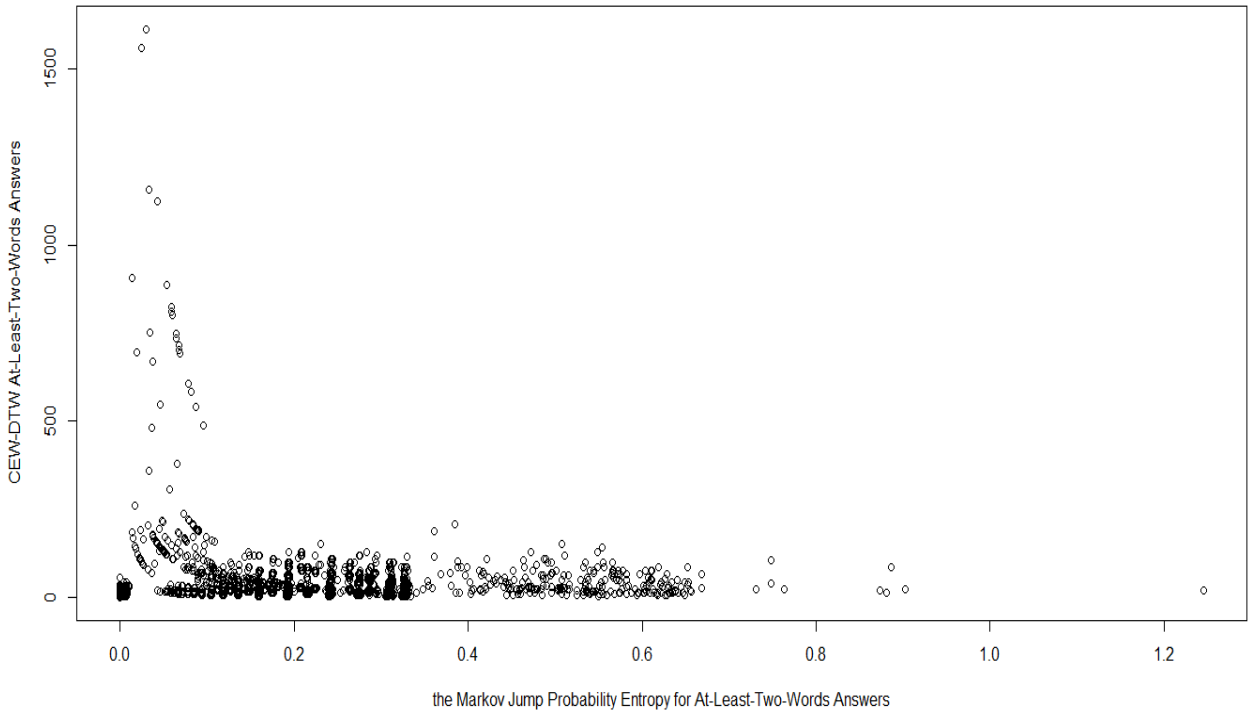


Figure 38: Relationship between CEW-DTW and the Markov Transition Probability Entropy

From Figure 38, CEW-DTW has a roughly negative relationship with $M_n(\boldsymbol{P})$. But, the structure of this negative relationship is not clear. But, the value of CEW-DTW and $M_n(\boldsymbol{P})$ of some answers are all very small. For the Markov Transition Probability Entropy, the high value of $M_n(\boldsymbol{P})$ for an answer also illustrates the higher quality of this answer. Therefore, sometimes, one of these methodologies can be used instead of another.

6.2.3 Relationship between the General Entropy and the Markov Transition Probability Entropy

The relationship between $E_n(\boldsymbol{P})$ and $M_n(\boldsymbol{P})$ is:

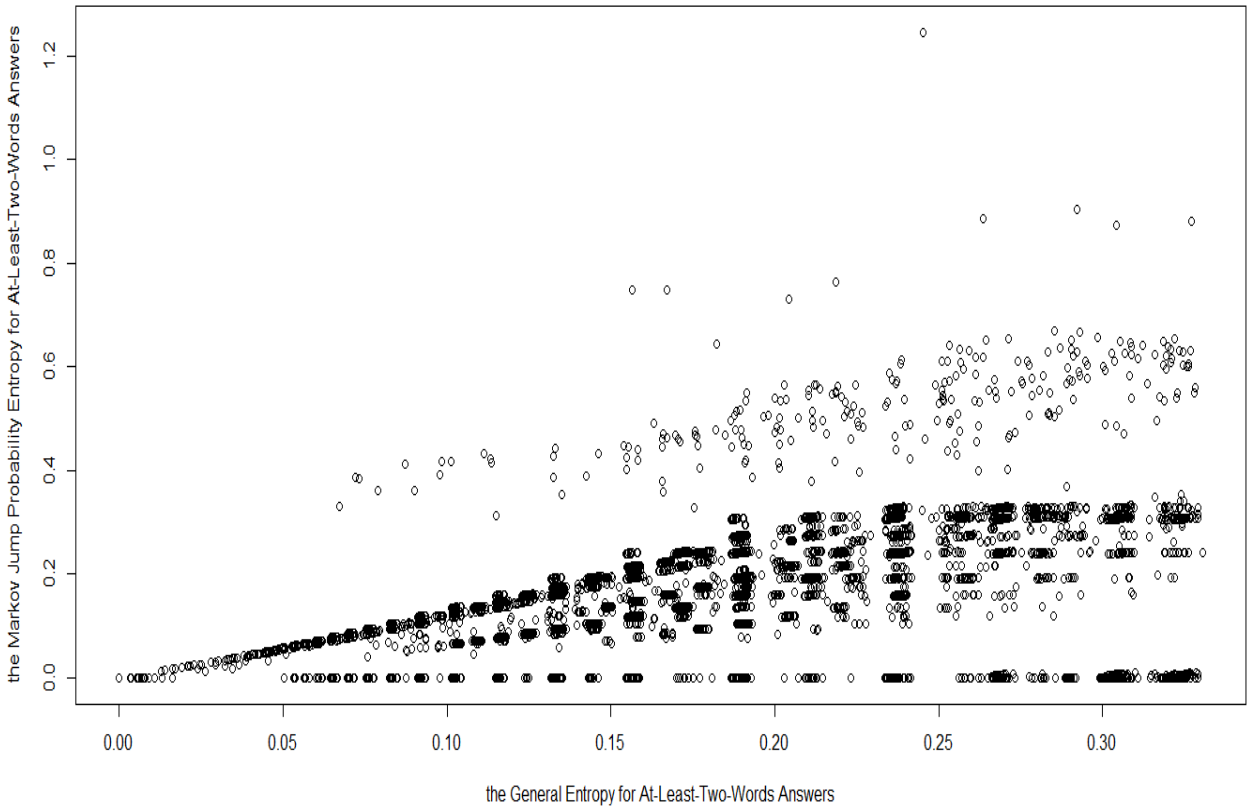


Figure 39: Relationship between the General Entropy and the Markov Transition Probability Entropy

From Figure 39, when $M_n(\boldsymbol{P})$ increases, $E_n(\boldsymbol{P})$ also increases. But, when $E_n(\boldsymbol{P})$ increases, $M_n(\boldsymbol{P})$ may not increase. So, they have one-way positive correlation. There are three distinct groups in this chart. In the first group, $M_n(\boldsymbol{P})$ is almost equal to zero. But, most $E_n(\boldsymbol{P})$ are not zero. $M_n(\boldsymbol{P})$ of the second group is between 0.0 and 0.25. $M_n(\boldsymbol{P})$ of the third group is larger than 0.3. The second group and the third group show a positive correlation between $M_n(\boldsymbol{P})$ and $E_n(\boldsymbol{P})$. $M_n(\boldsymbol{P})$ in the third group are larger than those in the second group. We choose some digitalized answers to analyze their features as following:

- Answer examples in the first group

No.	Typical Answer	Digitalized Vector
1	seat meet airline requirements	1,0,0,0
2	fit vista stroller	2,0,7
3	anyone know dimensions cot folded down weight trying avoid checking gracowondering weightsize travel ok	6,15,0,0,0,0,0,0,0,0,0,0,0,0
4	size sling extra large	14,0,0,0

When we get these digitalized vectors, we can use keywords to obtain matrixes about answers. For example, the transition probability matrix of No. 2 answer can be described as following:

	noise, fit, stroller
noise	0.0, 0.0, 1.0
fit	1.0, 0.0, 0.0
stroller	0.0, 0.0, 0.0

We compare $M_n(\boldsymbol{P})$ and $E_n(\boldsymbol{P})$ of answers as following:

No.	$E_n(\boldsymbol{P})$	$M_n(\boldsymbol{P})$
1	0.191	6.74e-14
2	0.187	6.74e-14

3	0.323	7.03e-14
4	0.116	6.66e-14

In this group, we find that when $E_n(\boldsymbol{P})$ is large, $M_n(\boldsymbol{P})$ may not be large. Therefore, $E_n(\boldsymbol{P})$ and $M_n(\boldsymbol{P})$ have no obvious correlation in this group.

Similarly, we can analyze some answer examples in the second and the third groups to compare their $M_n(\boldsymbol{P})$ and $E_n(\boldsymbol{P})$.

- Answer examples in the second group

No.1	Answer Content	keep strong year old cabinets special needs feels open cabinets dump whatever floor looking something prevent get cabinetsnnnn keep strong year old cabinets special needs feels open cabinets dump whatever floor looking something prevent himn nread morennn keep strong year old cabinets special needs feels open cabinets dump whatever floor looking something prevent get cabinetsn
	$E_n(\boldsymbol{P})$	0.076
	$M_n(\boldsymbol{P})$	0.087
No.2	Answer Content	locks cabinets love them work kitchen drawers
	$E_n(\boldsymbol{P})$	0.116
	$M_n(\boldsymbol{P})$	0.159
No.3	Answer Content	seat remove fit onto regular toilet

	$E_n(\boldsymbol{P})$	0.241
	$M_n(\boldsymbol{P})$	0.241

- Answer examples in the third group

No.1	Answer Content	anyone know naturepedic miniportable crib mattress fits crib looking organic mattress fits crib
	$E_n(\boldsymbol{P})$	0.312
	$M_n(\boldsymbol{P})$	0.641
No.2	Answer Content	mon old use nautilus car seat ive been reading reviews ppl stressing use car seat till child yrs old child defiently fits weight limit height reqirements know whats big deal mean year old likely weigh see problem anyone else recomend hold car seat till kid threennnn mon old use nautilus car seat ive been reading reviews ppl stressing use car seat till child yrs old child defiently fits nread morennn mon old use nautilus car seat ive been reading reviews ppl stressing use car seat till child yrs old child defiently fits weight limit height reqirements know whats big deal mean year old likely weigh see problem anyone else recomend hold car seat till kid threen
	$E_n(\boldsymbol{P})$	0.219
	$M_n(\boldsymbol{P})$	0.763

In the second group and the third group, we find that when $E_n(\boldsymbol{P})$ is large, $M_n(\boldsymbol{P})$ is also large. Therefore, $E_n(\boldsymbol{P})$ and $M_n(\boldsymbol{P})$ have correlation in this group. We choose three typical answers with similar high $E_n(\boldsymbol{P})$ but different $M_n(\boldsymbol{P})$ from three groups as following:

	Typical Answer	$E_n(\boldsymbol{P})$	$M_n(\boldsymbol{P})$
The First Group	anyone know dimensions cot folded down weight trying avoid checking gracowondering weightsize travel ok	0.323	7.03e-14
The Second Group	seat fit three across standard car seat like rxt	0.322	0.262
The Third Group	anyone know naturepedic miniportable crib mattress fits crib looking organic mattress fits crib	0.312	0.641

From above table, it is clear that answers in the second group or in the third group have more keywords than the answer in the first group. Furthermore, there are many keywords in the answer of the second group, keywords in the answer of the third group are separated by noises and distributed in the different locations. Subjectively, answers in the second group and the third group describes more details than the answer in the first group. In general, the answer in the third group is more reasonable and enables readers to get information thoroughly than answers in other groups.

- A Special Answer

We find a special point which high $M_n(\boldsymbol{P})$ as well as high $E_n(\boldsymbol{P})$. This answer is as following:

anyone use instead infant car seat first time mom like use advocate straight bir instead getting infant carseat anyone done experience been mind moving baby seat car big moving baby plus infant car seat heavy me advancennnn anyone use instead infant car seat first time mom like use advocate straight bir instead getting infant carseat anyone done hasn nread morennn

anyone use instead infant car seat first time mom like use advocate straight bir instead getting infant carseat anyone done experience been mind moving baby seat car big moving baby plus infant car seat heavy me advancen

The feature of this answer is that keywords frequently appear in this answer. These keywords are usually separated by noises. Compared with answers in the first and the second group, keywords in this answer are more related to each other.

By comparing three groups, we see that answers in the second group and the third group have more keywords than those in the first group. Though $M_n(\boldsymbol{P})$ of answers in the first group are almost zero, these answers still contain keywords. Therefore, we cannot say answers contain no information if $M_n(\boldsymbol{P})$ of answers are equal to zero. However, relative to answers in other groups, keywords information in answers of the first group is small. Thus, it demotes qualities of answers. Answers in the first group do not contain many keywords, which explains the lack of correlation between the two measurements. Keywords in answers in the second and third groups are separated by noises. Answers in these two groups contain more information than those in the first group and are more likely to be helpful answers. However, large values of $E_n(\boldsymbol{P})$ do not have an analogue in $M_n(\boldsymbol{P})$, therefore we cannot illustrate an obvious correlation between $M_n(\boldsymbol{P})$ and $E_n(\boldsymbol{P})$.

Though the relationship between $M_n(\boldsymbol{P})$ and $E_n(\boldsymbol{P})$ is not too clear, these methodologies can both be useful in different situations. For example, we can use these two methodologies when we want to judge the quality of interview answers. If we only check the key information of interview answers, $E_n(\boldsymbol{P})$, is the first choice since it mainly cares about keywords and noises. This situation usually appears in the group interview. Interviewees are usually given several minutes or a very short period. Thus, attendants usually narrate keywords to convey important information. Interviewers usually use these keywords to judge answer qualities. However, if we want to check not only key information but also the expression of language habit. $M_n(\boldsymbol{P})$ is recommended, since this model focuses on keywords as well as words' transitions. This situation usually appears in the one-by-one interview. In such kind of interview, the interviewer usually checks key information

of interviewees' answers and the expression format. Thus, the Markov Transition Probability Entropy, $M_n(\boldsymbol{P})$, is a good methodology to be used in this situation.

6.3 Wald–Wolfowitz runs test

Random patterns are important in statistics research. Since texts may contain keywords as well as noises, we can also analyze random patterns of texts. Wald–Wolfowitz runs test is a non-parametric method that is usually used to test random patterns of a data sequence.

6.3.1 Literature Review

Many studies have applied Wald–Wolfowitz runs test to verify patterns. In the original paper, Wald and Wolfowitz [117] develop a test method to verify the pattern of runs in terms of the total number of successes. Based on the Multidimensional Wald-Wolfowitz (MWW) runs test and the k-means clustering methodology, Leauhatong et al. [118] develop a new similarity methodology to verify images. Magel and Sasmito [119] compare the efficiency of simulation of the Wald-Wolfowitz test and the Kolmogorov-Smirnov test, when these two methodologies are applied in different situations. Mohanta et al. [120] develop a new scheme for shooting movies to assess the subshots within a shot and then for each subshot, their scheme is based on the Wald-Wolfowitz runs test. Song et al. [121] use the Wald–Wolfowitz runs test to verify the homogeneity of structural populations. George and Routray [122] use Multivariate Wald–Wolfowitz runs test to classify the data about eye movements. Kovačević et al. [123] adapt Wald–Wolfowitz run test to analyze data in terms of different soil environments. Their test results illustrate that different soil environment can be separated by the content of phenolic compounds. Chen et al. [124] use Wald–Wolfowitz runs test to analyze random patterns of signal noise. In their research, they adapt the median value of selected data to be a standard and use Wald–Wolfowitz runs test to analyze patterns. Song et al. [125] adapt Wald–Wolfowitz runs test to find the similarity between trace length and trace type. Wald–Wolfowitz runs test can also be used in non-normally-distribution data. Linkowska et al. [126] use this test to analyze mtDNA data in nontumor tissues. Since data

source are text data in our research, we can calculate the change rate between keywords and noises of an answer in terms of Wald–Wolfowitz runs test.

6.3.2 Wald–Wolfowitz Run Test

Wald–Wolfowitz Run test can be used to find the pattern of change frequency between keywords and noises. However, if the length of an answer is long, the value of the change frequency between keywords and noises may be large. To address this, we use the change rate between keywords and noises rather than the frequency to assess the pattern of an answer. Let $\boldsymbol{R}$ be the number of runs in this sequence and $\boldsymbol{N}$ be the total number of words. We can get the change rate as following:

$$\frac{\boldsymbol{R}}{\boldsymbol{N}}$$

In order to understand $\frac{\boldsymbol{R}}{\boldsymbol{N}}$, we illustrate it in an example as following. Given a 0/1 vector:

$$\{0,0,1,0,1,0,0\}$$

We can get $\boldsymbol{N} = 7$, since there are four changes in total. We obtain five parts in this vector: $\{0,0\}$, $\{1\}$, $\{0\}$, $\{1\}$, and $\{0,0\}$. So, $\boldsymbol{R}$=5. Therefore, we can get $\frac{\boldsymbol{R}}{\boldsymbol{N}} = \boldsymbol{5/7}$. In this chapter, we use 1 to represent keywords and 0 to represent noises. When we get $\frac{\boldsymbol{R}}{\boldsymbol{N}}$, our purpose is to compare it with CEW-DTW, $E_n(\boldsymbol{P})$, and $M_n(\boldsymbol{P})$ respectively to find their relationship. Here, we do not develop new methodology about $\frac{\boldsymbol{R}}{\boldsymbol{N}}$**.** We try to find whether the change of noise and keywords affect answer qualities by comparing $\frac{\boldsymbol{R}}{\boldsymbol{N}}$ with $E_n(P)$, $M_n(P)$, and CEW-DTW respectively. Since it is easy to calculate $\frac{\boldsymbol{R}}{\boldsymbol{N}}$, we try to find relationships between $\frac{\boldsymbol{R}}{\boldsymbol{N}}$ and other methodologies. Thus, we can judge whether we can use $\frac{\boldsymbol{R}}{\boldsymbol{N}}$ to replace other methodologies to assess answers.

6.3.3 Comparison between **R** and Length of answers, the General Entropy, the Markov Transition Probability Entropy, and CEW-DTW respectively

The number of runs, $\boldsymbol{R}$, is used mainly to analyze answers from the viewpoint of patterns' properties. It is easy to obtain the value of $\boldsymbol{R}$, so, we try to compare $\boldsymbol{R}$ with different methodologies before we compare $\frac{\boldsymbol{R}}{\boldsymbol{N}}$ and other methodologies:

- Compare **R** and Length of answers

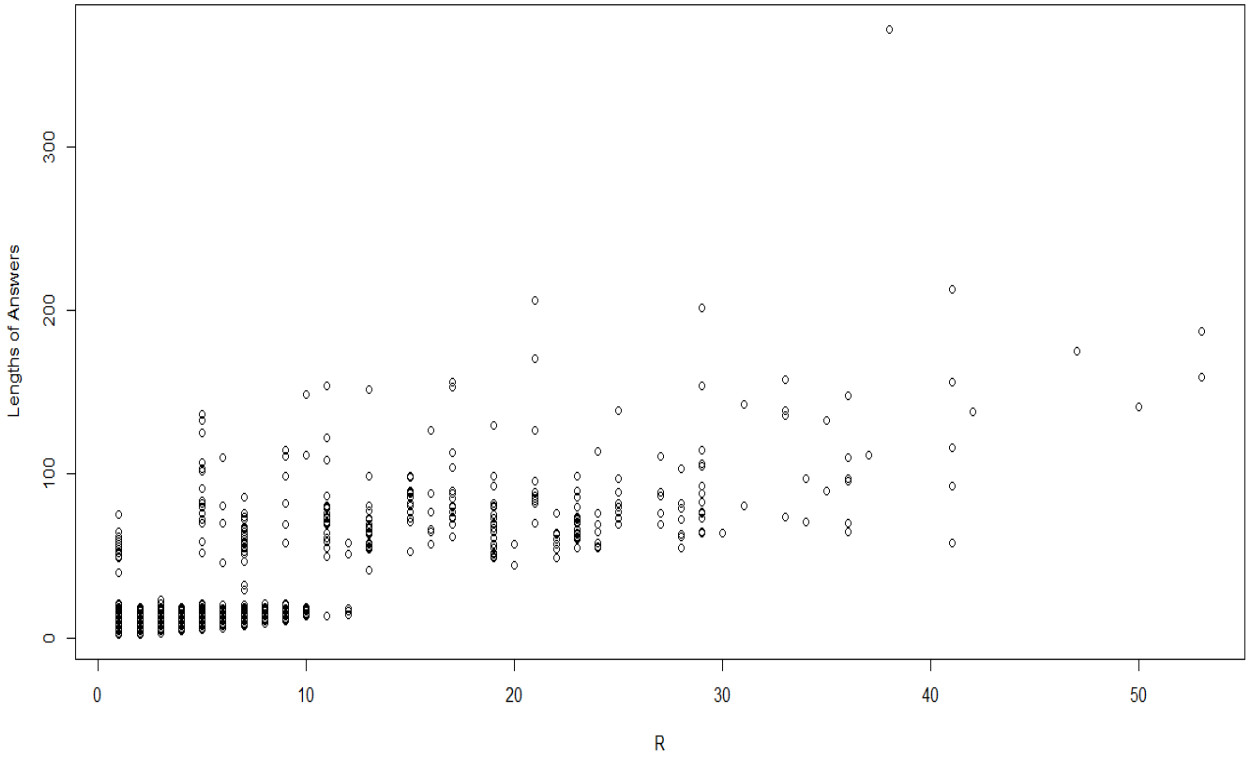


Figure 40: Relationship between R and Lengths of answers

From Figure 40, we find that $\boldsymbol{R}$ has a roughly positive relationship with lengths of answers. It clearly illustrates that the value of $\boldsymbol{R}$ is larger when the length of an answer becomes longer.

- Compare **R** and the General Entropy

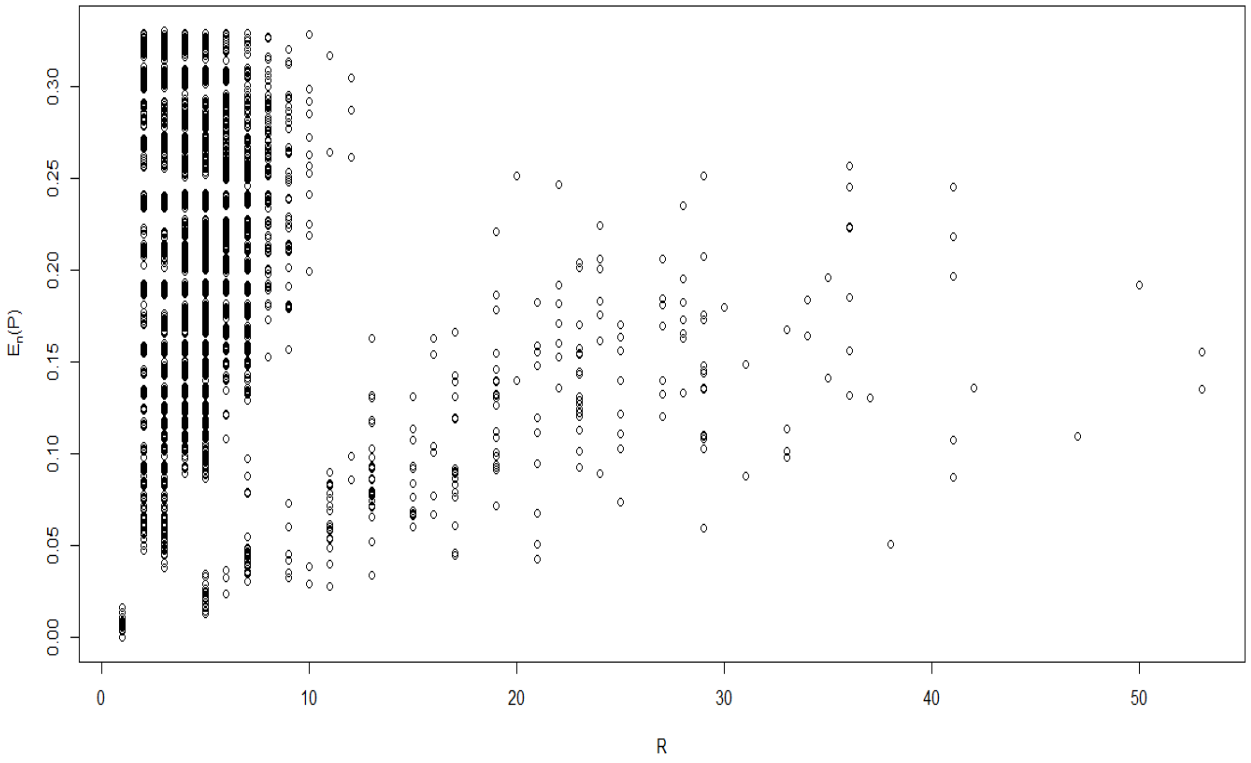


Figure 41: Relationship between R and the general entropy

The Figure 41 shows two groups. In the first group, most values of $\boldsymbol{R}$ are less than 10. There is no obvious relationship between $\boldsymbol{R}$ and $E_n(P)$ in this group. However, the second group shows a positive relationship between $\boldsymbol{R}$ and $E_n(P)$. Thus, there is not a clear relationship between $\boldsymbol{R}$ and $E_n(P)$.

- Compare **R** and the Markov Transition Probability Entropy

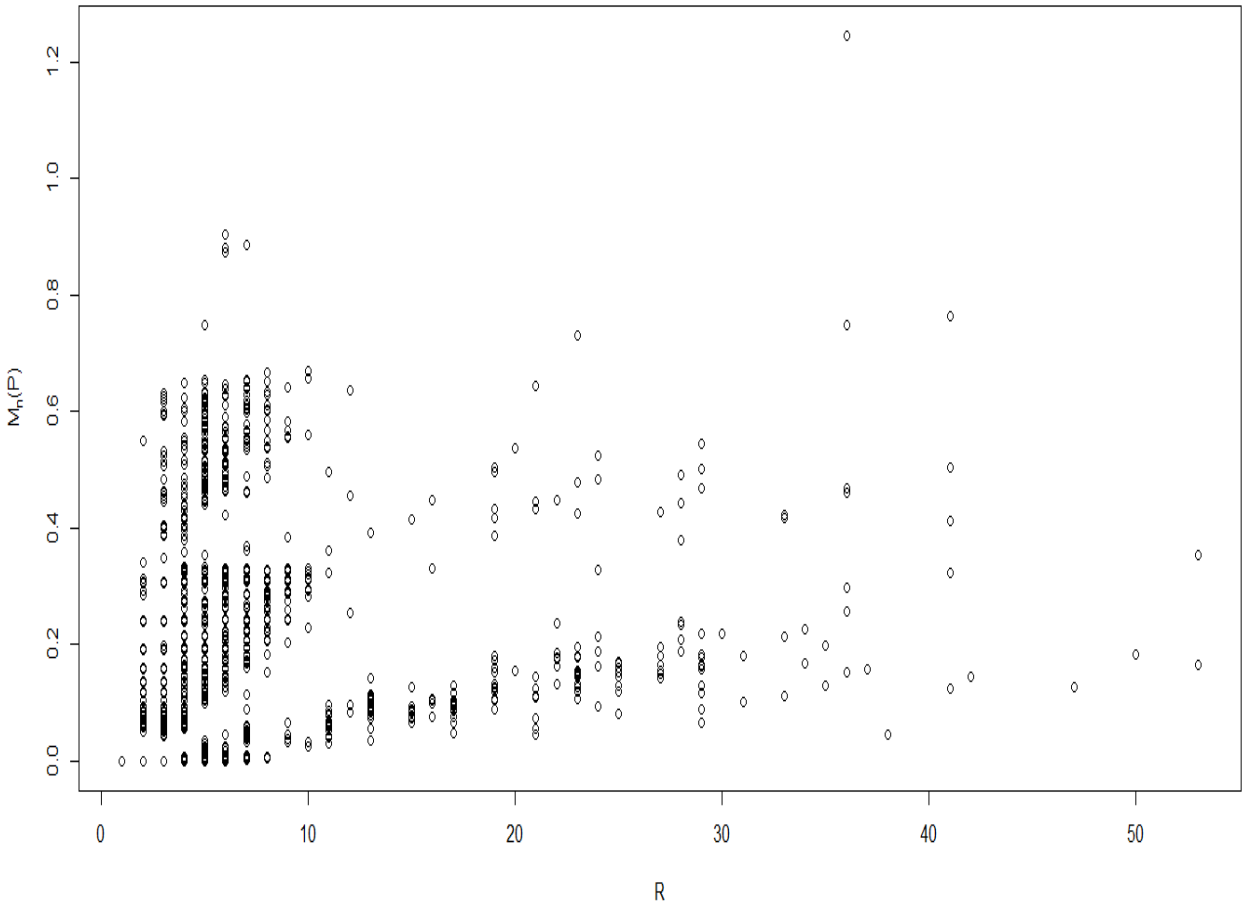


Figure 42: Relationship between R and the Markov Transition Probability Entropy

In Figure 42, there are also two groups. In the first group, most values of ***R*** are also less than 10. Also, there is no obvious relationship between ***R*** and $M_n(P)$ in this group. However, the second group also shows a positive relationship between ***R*** and $M_n(P)$. Therefore, we also cannot conclude the relationship between ***R*** and $M_n(P)$.

- Compare **R** and CEW-DTW

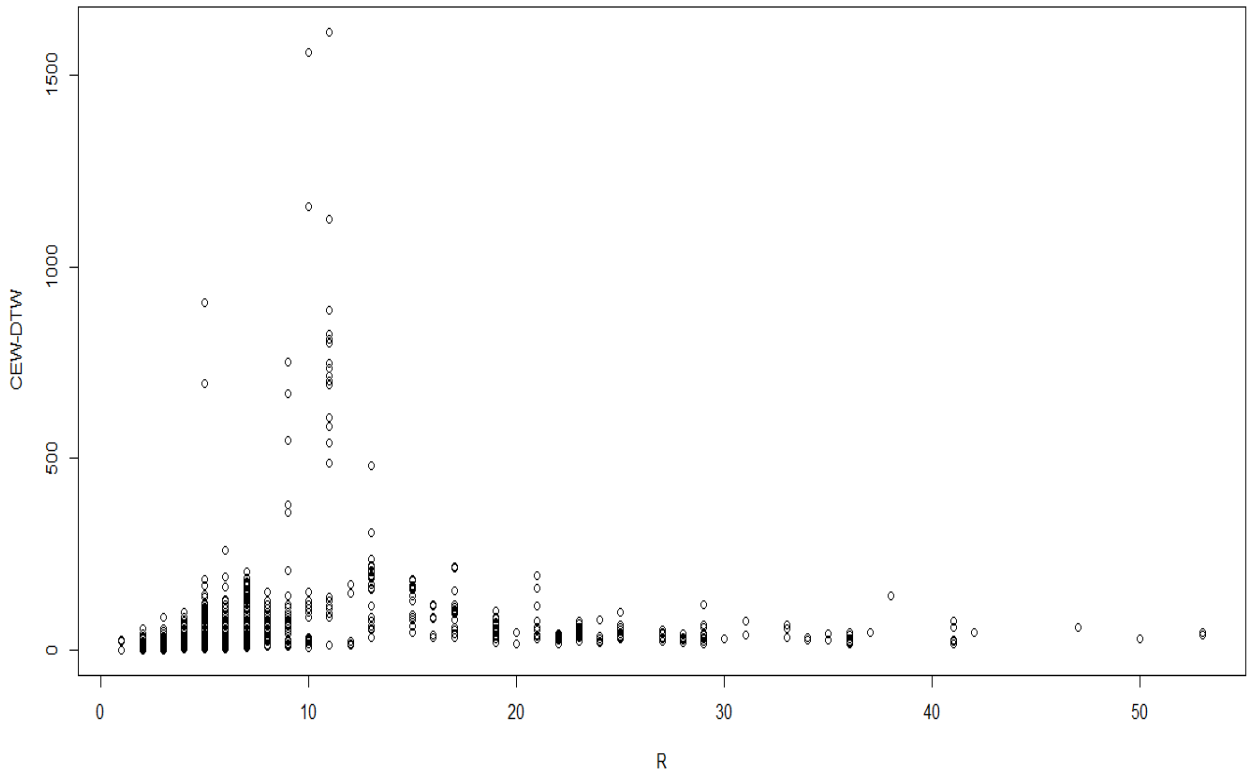


Figure 43: Relationship between R and CEW-DTW

In Figure 43, there is no clear relationship between $\boldsymbol{R}$ and CEW-DTW. The point with the largest value of CEW-DTW is corresponding to the value of $\boldsymbol{R}$, 11. Thus, there is also not a clear relationship between $\boldsymbol{R}$ and CEW-DTW to test the quality of answers.

From Figures 40, 41, 42, and 43, we find that though there is a relationship between $\boldsymbol{R}$ and lengths of answers, there are no obvious relationships between $\boldsymbol{R}$ and $E_n(P)$, $M_n(P)$, CEW-DTW respectively. Thus, we cannot use $\boldsymbol{R}$ to roughly replace $E_n(P)$, $M_n(P)$, and CEW-DTW to test qualities of answers.

6.3.4 Comparison R/N and the General Entropy, the Markov Transition Probability Entropy, and CEW-DTW respectively

Since $\frac{R}{N}$ can also be calculated easily, we try to compare $\frac{R}{N}$ with $E_n(P)$, CEW-DTW, and $M_n(P)$ respectively to see whether $\frac{R}{N}$ can replace these methodologies respectively. We firstly show the density distribution of $\frac{R}{N}$ for these answers:

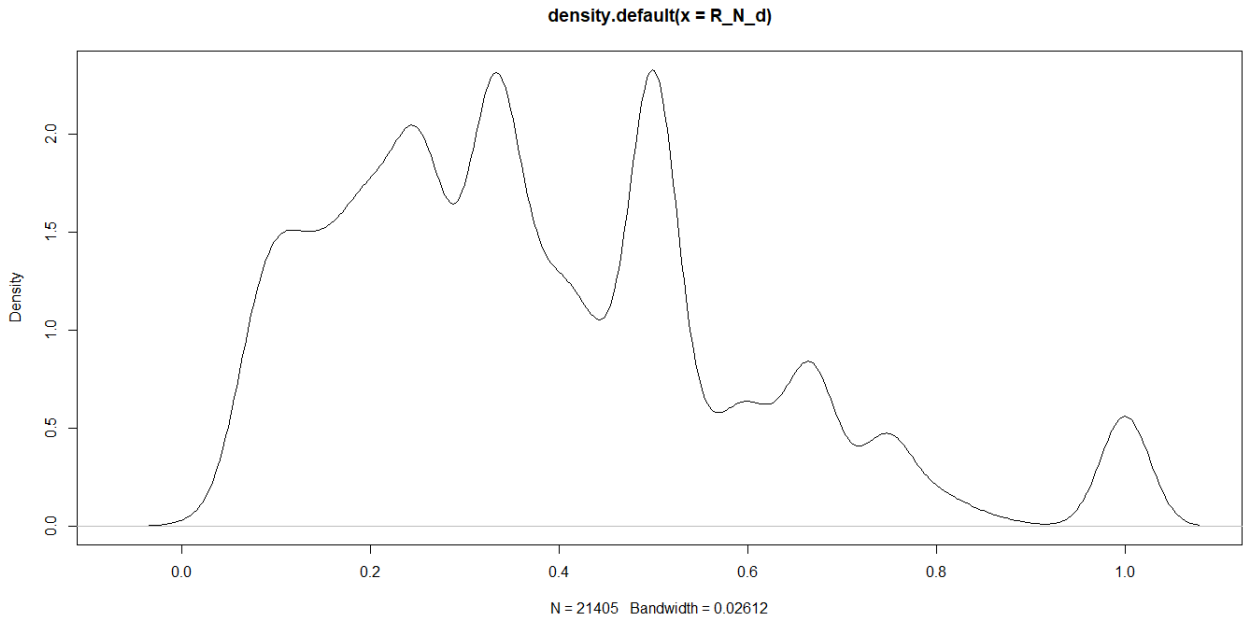


Figure 44: Density distribution of R/N

From Figure 44, we can see that the distribution of $\frac{R}{N}$ is not normal. $\frac{R}{N}$ of most answers are between 0.05 and 0.8. But there are still some answers, $\frac{R}{N}$ of which is equal to 1. Length of such an answer is equal to runs of this answer. For example, if an answer contains two words, one word is the noise, another word is a keyword. $\frac{R}{N}$ of this answer is equal to 1.

- Compare **R/N** and CEW-DTW

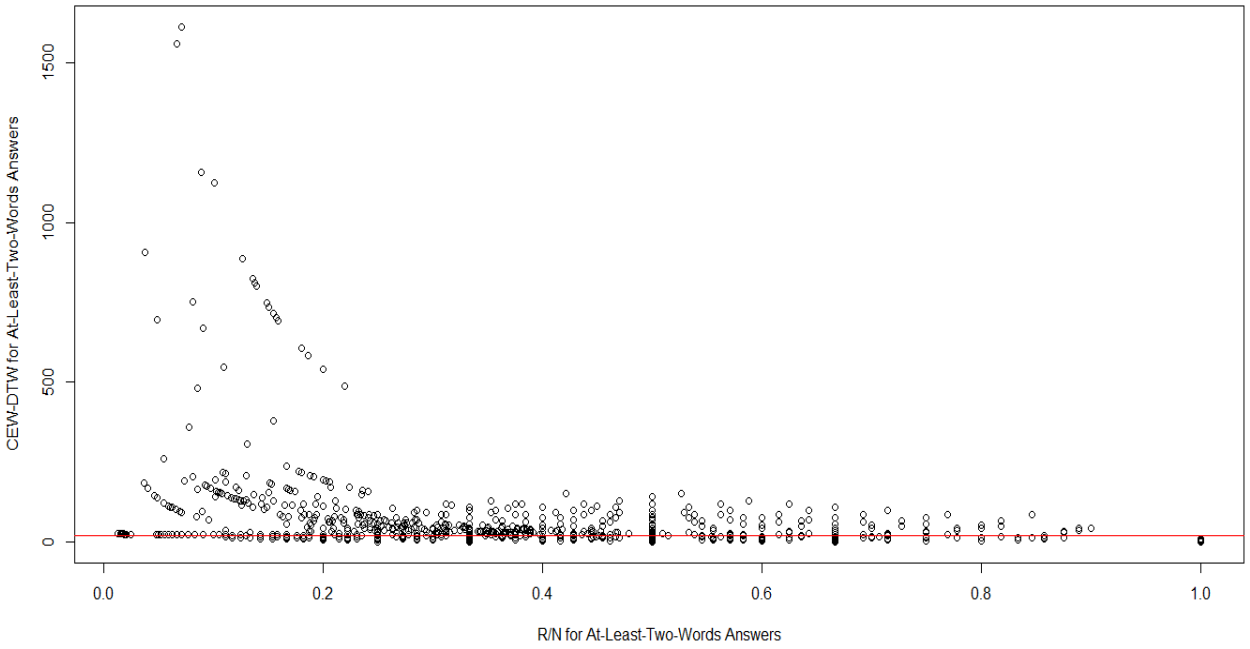


Figure 45: Relationship between R/N and CEW-DTW

- Compare R/N and the General Entropy

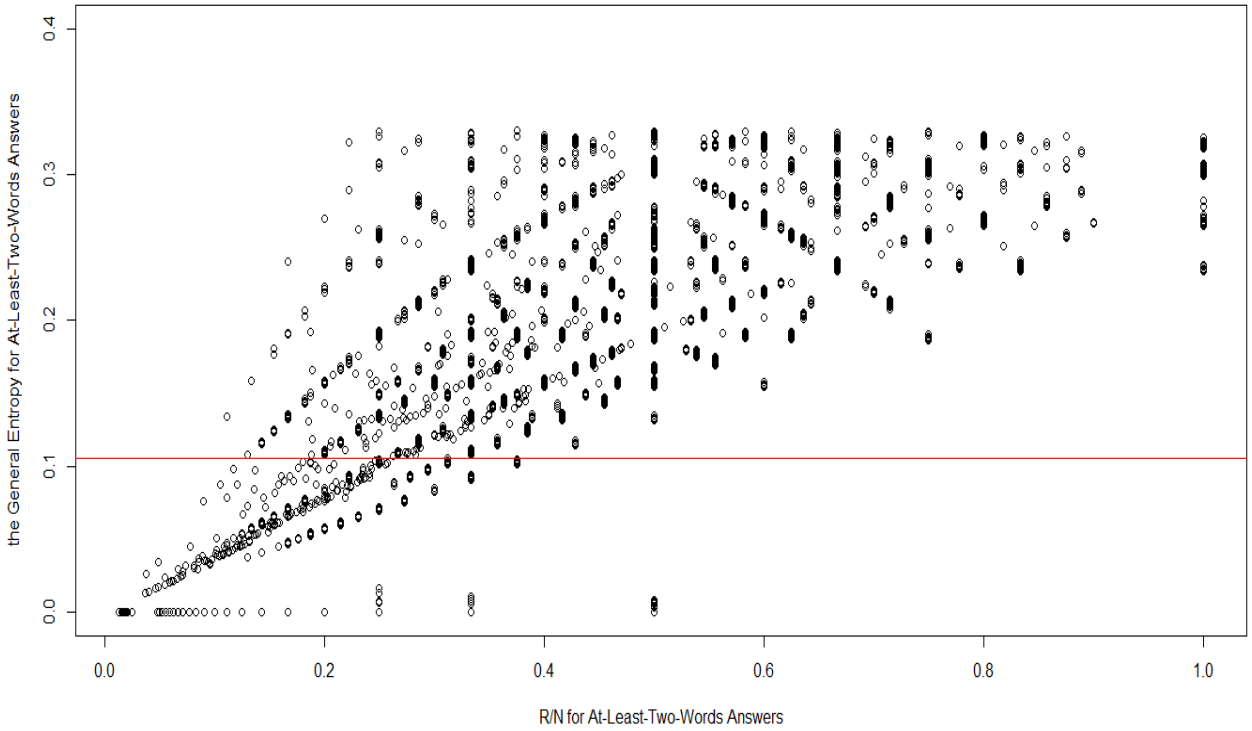


Figure 46: Relationship between R/N and the General Entropy

- Compare R/N and the Markov Transition Probability Entropy

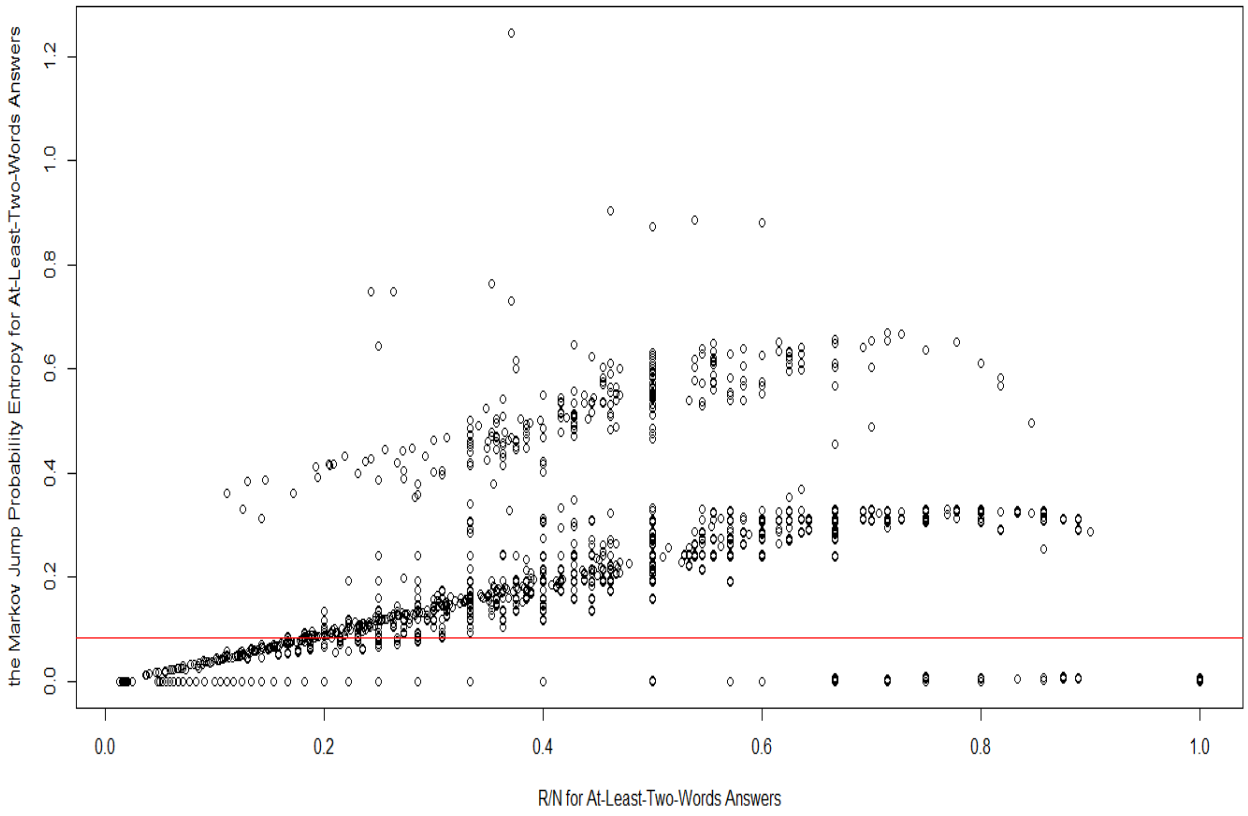


Figure 47: Relationship between R/N and the Markov Transition Probability Entropy

From Figures 45, 46 and 47, we find that CEW-DTW shows a roughly negative relationship with $\frac{R}{N}$. It illustrates that when an answer has high random patterns, this answer is also close to the "ideal" answer. In practice, if an answer contains many keywords, the quality of this answer is high. However, relationships between $\frac{R}{N}$ and $M_n(P)$ or $E_n(P)$ are not obviously positive. Mean values of $M_n(P)$, $E_n(P)$, and CEW-DTW are represented by the red line in each figure. We find that these mean values are close to 0. If an answer has a high value of random pattern, we cannot judge the range of value of $M_n(P)$ or $E_n(P)$. Therefore, $\frac{R}{N}$ has no obvious relationship with $M_n(P)$ or $E_n(P)$. Therefore, we conclude that $\frac{R}{N}$ can be applied to replace CEW-DTW to assess answers roughly. But, $\frac{R}{N}$ cannot be used to replace $M_n(P)$ or $E_n(P)$ to assess answers.

6.4 Comparison Lengths of Answers with the General Entropy, the Markov Transition Probability Entropy, and CEW-DTW respectively

Length of an answer is another statistical property. We can also compare length of answers and different methodologies so as to check whether length of answers can replace these methodologies to assess answers.

- Compare Length of answers and the Markov Transition Probability Entropy

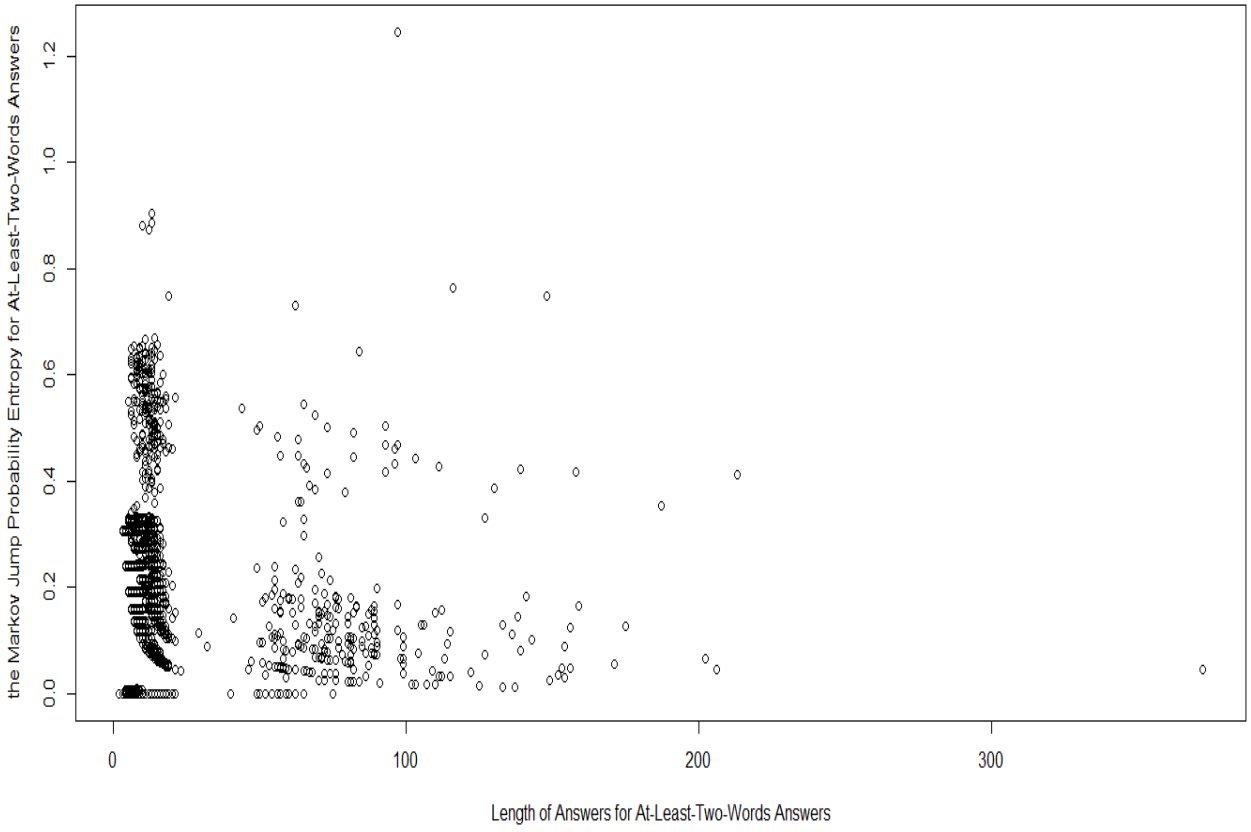


Figure 48: Relationship between Lengths and the Markov Transition Probability Entropy

- Compare Length of answers and the General Entropy

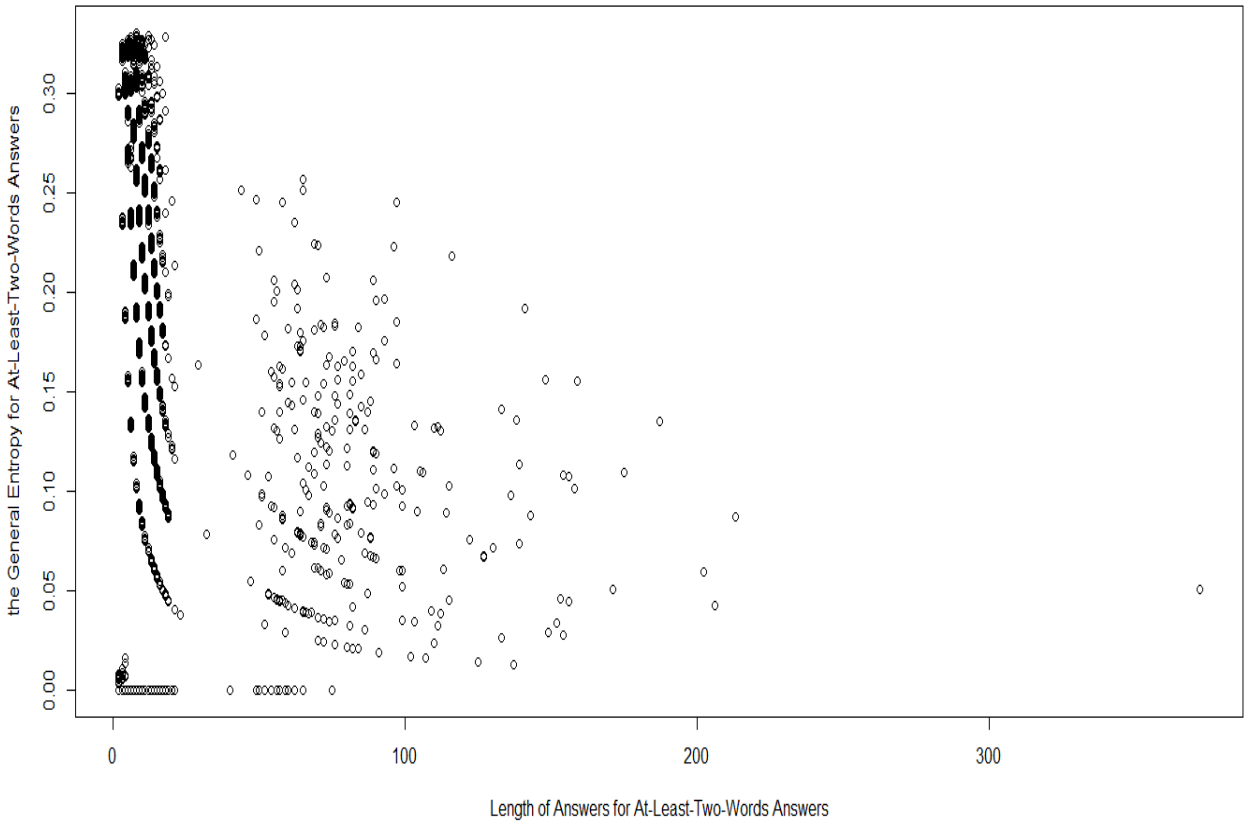


Figure 49: Relationship between Lengths and the General Entropy

- Compare Length of answers and CEW-DTW

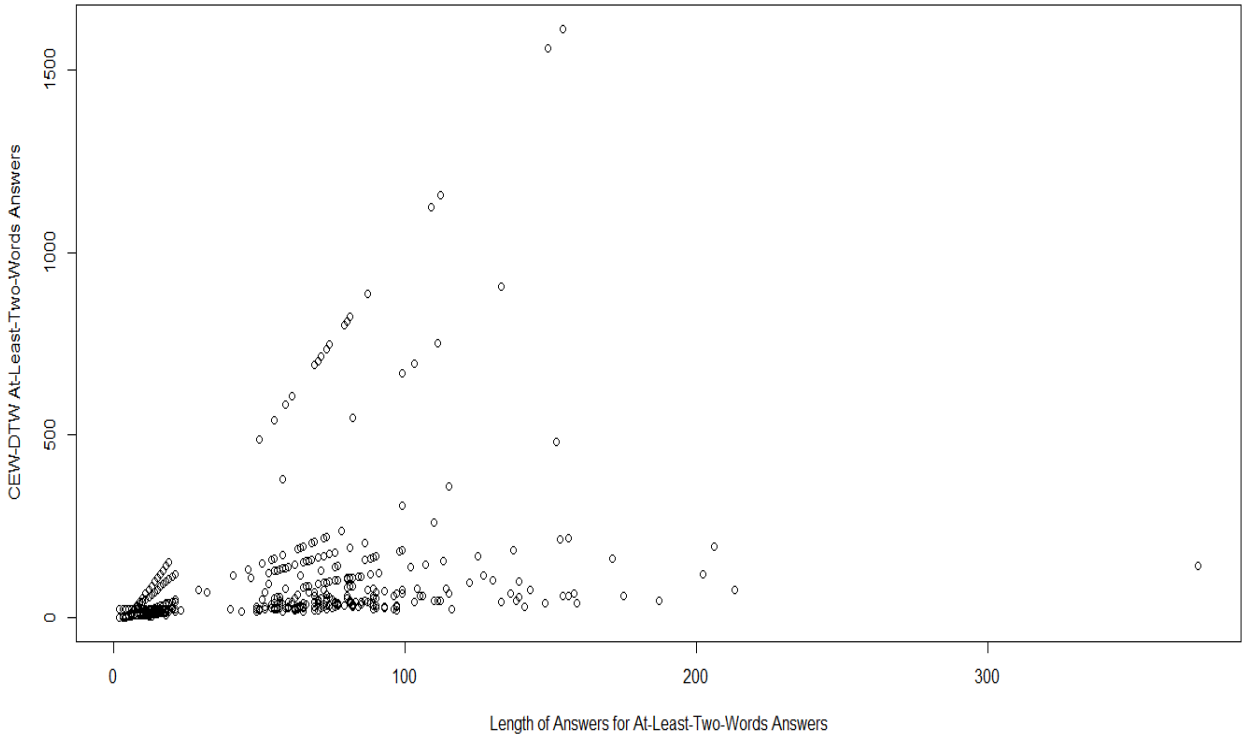


Figure 50: Relationship between Lengths and CEW-DTW

From Figures 48, 49 and 50, we find that each relationship can be separated to three groups. Lengths of answers cannot give more information to judge $E_n(\boldsymbol{P})$ or $M_n(\boldsymbol{P})$. However, lengths of answers show a penalized relationship with CEW-DTW. That is, we cannot conclude that answers with long or short lengths have high or low value of CEW-DTW. If an answer has a median length, the length has a positive relationship with CEW-DTW. Therefore, length of answers cannot be adapted to replace $E_n(\boldsymbol{P})$ or $M_n(\boldsymbol{P})$ to assess answers in general. But length of answers can replace CEW-DTW to assess answers if lengths of answers are not too long or short.

6.5 Conclusion

From above statements of comparison, we can make some conclusions. $E_n(\boldsymbol{P})$, $M_n(\boldsymbol{P})$, or CEW-DTW can be applied in different situations. $E_n(\boldsymbol{P})$ fit for long or short answers. $M_n(\boldsymbol{P})$ fits for long answers. CEW-DTW uses an "ideal" answer as a standard to rank answers. If we hope to care

about keywords as well as time series sequence of keywords, we can use CEW-DTW to judge answer qualities. If we use $E_n(\boldsymbol{P})$ and $M_n(\boldsymbol{P})$ as an assessment, we do not need a standard to judge answer qualities. In real answers, noises are also necessary and cannot be discarded. If an answer only has keywords without noises, it cannot easily to be considered as a good or bad answer. Thus, if we hope to verify answers' quality in terms of keywords and noises together, we can use $E_n(\boldsymbol{P})$ to assess answers. When we analyze answers, we are usually given multiple keywords for analysis. If we hope to find inner-connection of keywords and noises, we can use $M_n(\boldsymbol{P})$ to assess answers. Therefore, $E_n(\boldsymbol{P})$, $M_n(\boldsymbol{P})$, or CEW-DTW can be applied in various situations.

R/N has a roughly negative relationship with CEW-DTW and less an obvious relationship with $E_n(\boldsymbol{P})$ or $M_n(\boldsymbol{P})$. Thus, we can sometimes use R/N to replace CEW-DTW, since R/N is easily to be computed. In the future, we plan to develop some mew methodologies in terms of Wald–Wolfowitz Run test. We can then try to verify answer qualities from the viewpoint of random patterns.

Though length of answers has no necessary relationship with $E_n(\boldsymbol{P})$ and $M_n(\boldsymbol{P})$, it has penalized relationship with CEW-DTW. Thus, if the length of an answer is not too long or short, the length of an answer has a rough positive relationship with CEW-DTW. In the future, we plan to analyze how long of an answer can be assessed by CEW-DTW.

Chapter 7

7 Conclusion and Future Plan

7.1 Conclusion

Throughout the entire process of the thesis, we summarize main contributions in several parts. Firstly, we develop CEW-DTW. This methodology gives us a standard—an "ideal" answer—to rank answers. It has been proved to have a better ranking performance than Dynamic Time Warping and Dynamic Time Warping-Delta. Secondly, we develop KL-CEW-DTW from CEW-DTW. This methodology rank answers from the viewpoint of distributions of keywords and noise. It is proven to be better than CEW-DTW in ranking performance. Thirdly, we develop the general entropy, which use probabilities of noise and keywords to analyze answers. We develop an imaginary answer with the maximum entropy probabilities from the global probabilities in terms of the general entropy methodology. The maximum general entropy answer gives us a way to judge which keywords are important. We also find a way to determine the optimum number of keywords. According to this optimum number, we do not need to select too many keywords. Fourthly, we study inner connections of noise and keywords by applying the Markov transition matrix. This methodology contributes to judge which two keywords are usually connected. The inner connections are helpful to find the trend of speech.

Another contribution is that we can regard CEW-DTW, KL-CEW-DTW, the General Entropy, and the Transition Probability Entropy together as a simple development process of Artificial Intelligence from Semi-Supervised Learning to Supervised Learning. For the large volume of answers, our analysis process is from the simple analysis stage to the complicated analysis stage. We begin by explaining Unsupervised Learning, Semi-Supervised Learning, and Supervised Learning. According to the description of some literatures ([127], [128]), these three learnings can be explained as follows:

- Unsupervised Learning: the data set is unlabeled
- Supervised Learning: the data set is labeled

- Semi-Supervised Learning: this learning is between Unsupervised Learning and Supervised Learning. It means some data are labeled.

Therefore, if the set of answers is unlabeled, we can use unsupervised learning to analyze these answers. Here, we consider unlabeled answers to be answers with no keywords. Similarly, if we use different keywords to analyze answers, we can apply supervised learning to analyze them; if we regard different keywords as the same keyword to analyze answers, we can apply semi-supervised learning to analyze them. Semi-supervised learning can enhance efficiency of assignments which are ever carried out by supervised learning, when the volume of labeled data is very large ([129]). By applying unlabeled data, some supervised methodologies can be transferred to semi-supervised methodologies ([130]). So, semi-supervised learning may perform as well as supervised learning, but with some performance difference ([129]). When we use semi-supervised and supervised learning to analyze data respectively, analysis results may roughly similar. Since our analysis about answers are related to keywords and the noise, we can resolve these analysis into fields of semi-supervised learning and supervised learning respectively. Frthermore, different supervised learning methodologies may perform significant variability across the problems, it means excellent methodologies sometimes show bad performances, and poor efficient methodologies sometimes show wonderful performances ([131]).

For CEW-DTW and KL-CEW-DTW, we use the number 1 and 0 to represent the keyword and the noise respectively in an answer. Since 1 and 0 represents any keyword and noise in these two methodologies, CEW-DTW and KL-CEW-DTW can be regarded as two methodologies of semi-supervised learning. For the general entropy and the transition probability entropy, we use numbers: $0,1,2,\ldots,n$ to represent the noise, the keyword 1, the keyword 2, …, and the keyword n respectively in an answer. Thus, these two methodologies belong to supervised learning, but not strictly supervised. Since supervised learning is more complicated than semi-supervised learning, our data analysis for answers starts from semi-supervised learning. Therefore, we firstly develop CEW-DTW and KL-CEW-DTW, then we develop the general entropy and the transition probability entropy. The comparison results in Chapter 6 illustrate that performances of some methodologies are roughly similar indeed. On the other hand, though we consider both the general

entropy and the transition probability entropy to be methodologies about semi-supervised learning, their performances indeed show some difference in ranking answers. Based on above analysis, we believe that our research progress can be described as a simple development process of Artificial Intelligence. Currently, our developed methodologies are not in consideration of grammar. These methodologies are mainly based on qualities of keywords and the noise. Though there are some methodologies or systems, which care about linguistic properties of texts ([132], [133]), these methodologies or systems are required to be supported by powerful capabilities of computing. So, our methodologies are not complete artificial intelligent methodologies. However, our methodologies can be developed better to combine linguistic grammar in the future.

These methodologies can be applied in many fields. CEW-DTW or KL-DTW-CEW is developed from DTW. Since DTW is also widely applied in image analysis. We can also adapt CEW-DTW or KL-CEW-DTW to analyze images. We plan to use these methodologies to analyze features of image edges. Therefore, we can do some researches about image classification or clustering. We also try to introduce the General Entropy and the Markov Transition Probability Entropy to human resource managers to help them assess the interview quality. These methodologies can be applied in different situations. For example, if human resource managers hope to assess introduction qualities of various interviewees, the Markov Transition Probability Entropy will be helpful. Since introduction is usually long and the Markov Transition Probability Entropy can check the inner-connection of words, human resource managers can verify the logicality of introduction. If human resource managers want to check answer qualities of different interviewees, the General Entropy is helpful, since answers are usually verified in terms of key information as well as other useless words.

7.2 Future Plan

First of all, we plan to continue analyzing the Markov transition probability entropy, we try to analyze entropy of multiple transitions (e.g. more than two keywords). Secondly, since we combine Java and R together to implement methodologies, we try to use these methodologies to

deal with more large computation by applying distributed computational methodologies. Also, we want to develop R packages to implement functions of Java codes in the future. For example, we can develop the R package to calculate the transition matrix. Thus, the Markov Transition Probability Entropy can be completely implemented by R. Furthermore, statistical methodologies are hard to be explained clearly in practice. We plan to adapt J2EE technology to develop a platform. On this platform, we can implement various dynamic data visualization to explain data. It can help readers to understand our methodologies intuitively.

References

[1]. Ikonomakis, M., Sotiris Kotsiantis, and V. Tampakas. "Text classification using machine learning techniques." WSEAS transactions on computers 4, no. 8 (2005): 966-974.

[2]. Onan, Aytuğ, and Serdar Korukoğlu. "A feature selection model based on genetic rank aggregation for text sentiment classification." Journal of Information Science 43, no. 1 (2017): 25-38.

[3]. Liu, Pengfei, Xipeng Qiu, and Xuanjing Huang. "Adversarial multi-task learning for text classification." arXiv preprint arXiv:1704.05742 (2017).

[4]. Xuan, Jifeng, He Jiang, Zhilei Ren, Jun Yan, and Zhongxuan Luo. "Automatic bug triage using semi-supervised text classification." arXiv preprint arXiv:1704.04769 (2017).

[5]. Xu, Shuo. "Bayesian Naïve Bayes classifiers to text classification." Journal of Information Science 44, no. 1 (2018): 48-59.

[6]. Chen, Jiangning, Heinrich Matzinger, Haoyan Zhai, and Mi Zhou. "Centroid estimation based on symmetric KL divergence for Multinomial text classification problem." arXiv preprint arXiv:1808.10261 (2018).

[7]. Garg, Sahaj, Vincent Perot, Nicole Limtiaco, Ankur Taly, Ed H. Chi, and Alex Beutel. "Counterfactual Fairness in Text Classification through Robustness." arXiv preprint arXiv:1809.10610 (2018).

[8]. Shu, Lei, Hu Xu, and Bing Liu. "Doc: Deep open classification of text documents." arXiv preprint arXiv:1709.08716 (2017).

[9]. Yogatama, Dani, Chris Dyer, Wang Ling, and Phil Blunsom. "Generative and discriminative text classification with recurrent neural networks." arXiv preprint arXiv:1703.01898(2017).

[10]. Ive, Julia, George Gkotsis, Rina Dutta, Robert Stewart, and Sumithra Velupillai. "Hierarchical neural model with attention mechanisms for the classification of social media text related to mental health." In Proceedings of the Fifth Workshop on Computational Linguistics and Clinical Psychology: From Keyboard to Clinic, pp. 69-77. 2018.

[11]. Li, Yan, and Jieping Ye. "Learning Adversarial Networks for Semi-Supervised Text Classification via Policy Gradient." In Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp. 1715-1723. ACM, 2018.

[12]. Liu, Ming, Gholamreza Haffari, Wray Buntine, and Michelle Ananda-Rajah. "Leveraging linguistic resources for improving neural text classification." In Proceedings of the Australasian Language Technology Association Workshop 2017, pp. 34-42. 2017.

[13]. Saha, Avijit, Vishal Kakkar, and T. Ravindra Babu. "Noise-aware Missing Shipment Return Comment Classification in E-Commerce." (2018).

[14]. Abualigah, Laith Mohammad, Ahamad Tajudin Khader, and Essam Said Hanandeh. "A new feature selection method to improve the document clustering using particle swarm optimization algorithm." Journal of Computational Science 25 (2018): 456-466.

[15]. Xu, Jingyun, Yi Cai, Shuai Wang, Kai Yang, Qing Du, Jun Zhang, Li Yao, and Jingjing Li. "A Text Clustering Algorithm to Detect Basic Level Categories in Texts." In International Conference on Web-Based Learning, pp. 72-81. Springer, Cham, 2017.

[16]. Mohammed, Athraa Jasim, Yuhanis Yusof, and Husniza Husni. "Fireflyclust: an automated hierarchical text clustering approach." Jurnal Teknologi 79, no. 5 (2017): 11-22.

[17]. Grieco, Antonio, Massimo Pacella, and Marzia Blaco. "On the application of text clustering in Engineering Change process." Procedia CIRP 62 (2017): 187-192.

[18]. Xu, Jiaming, Bo Xu, Peng Wang, Suncong Zheng, Guanhua Tian, and Jun Zhao. "Self-taught convolutional neural networks for short text clustering." Neural Networks 88 (2017): 22-31.

[19]. Dörpinghaus, Jens, Sebastian Schaaf, and Marc Jacobs. "Soft document clustering using a novel graph covering approach." BioData Mining 11, no. 1 (2018): 11.

[20]. Matei, Liviu Sebastian, and Stefan Trausan-Matu. "TEXT CLUSTERING BY AUTHOR USING THE TIME SERIES MODEL." UNIVERSITY POLITEHNICA OF BUCHAREST SCIENTIFIC BULLETIN SERIES C-ELECTRICAL ENGINEERING AND COMPUTER SCIENCE 80, no. 1 (2018): 3-14.

[21]. Abualigah, Laith Mohammad, Ahamad Tajudin Khader, Mohammed Azmi Al-Betar, and Osama Ahmad Alomari. "Text feature selection with a robust weight scheme and dynamic dimension reduction to text document clustering." Expert Systems with Applications 84 (2017): 24-36.

[22]. Abualigah, Laith Mohammad, and Ahamad Tajudin Khader. "Unsupervised text feature selection technique based on hybrid particle swarm optimization algorithm with genetic operators for the text clustering." The Journal of Supercomputing 73, no. 11 (2017): 4773-4795.

[23]. Lu, Huimin, Baofeng Guo, Juntao Liu, and Xijun Yan. "A shadow removal method for tesseract text recognition." In Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI), 2017 10th International Congress on, pp. 1-5. IEEE, 2017.

[24]. Tian, Shu, Xu-Cheng Yin, Ya Su, and Hong-Wei Hao. "A unified framework for tracking based text detection and recognition from web videos." IEEE transactions on pattern analysis and machine intelligence 40, no. 3 (2018): 542-554.

[25]. Yang, Chun, Xu-Cheng Yin, Zejun Li, Jianwei Wu, Chunchao Guo, Hongfa Wang, and Lei Xiao. "AdaDNNs: Adaptive Ensemble of Deep Neural Networks for Scene Text Recognition." arXiv preprint arXiv:1710.03425 (2017).

[26]. Shi, Baoguang, Xiang Bai, and Cong Yao. "An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition." IEEE transactions on pattern analysis and machine intelligence 39, no. 11 (2017): 2298-2304.

[27]. Bušta, Michal, Lukáš Neumann, and Jirı Matas. "Deep textspotter: An end-to-end trainable scene text localization and recognition framework." In Computer Vision (ICCV), 2017 IEEE International Conference on, pp. 2223-2231. IEEE, 2017.

[28]. Xie, Zecheng, Zenghui Sun, Lianwen Jin, Hao Ni, and Terry Lyons. "Learning spatial-semantic context with fully convolutional recurrent network for online handwritten Chinese text recognition." IEEE transactions on pattern analysis and machine intelligence 40, no. 8 (2018): 1903-1917.

[29]. Liu, Zichuan, YIxing Li, Fengbo Ren, Wang Ling Goh, and Hao Yu. "SqueezedText: A Real-Time Scene Text Recognition by Binary Convolutional Encoder-Decoder Network." In AAAI. 2018.

[30]. Liao, Minghui, Baoguang Shi, Xiang Bai, Xinggang Wang, and Wenyu Liu. "TextBoxes: A Fast Text Detector with a Single Deep Neural Network." In AAAI, pp. 4161-4167. 2017.

[31]. Raifer, Nimrod, Fiana Raiber, Moshe Tennenholtz, and Oren Kurland. "Information Retrieval Meets Game Theory: The Ranking Competition Between Documents? Authors." In Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 465-474. ACM, 2017.

[32]. Xiong, Chenyan, Zhengzhong Liu, Jamie Callan, and Eduard Hovy. "JointSem: Combining Query Entity Linking and Entity based Document Ranking." In Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, pp. 2391-2394. ACM, 2017.

[33]. Pandey, Gaurav, Zhaochun Ren, Shuaiqiang Wang, Jari Veijalainen, and Maarten de Rijke. "Linear feature extraction for ranking." Information Retrieval Journal (2018): 1-26.

[34]. Wang, Chengyu, Guomin Zhou, Xiaofeng He, and Aoying Zhou. "NERank+: a graph-based approach for entity ranking in document collections." Frontiers of Computer Science 12, no. 3 (2018): 504-517.

[35]. Wei, Zeng, Jun Xu, Yanyan Lan, Jiafeng Guo, and Xueqi Cheng. "Reinforcement learning to rank with Markov decision process." In Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 945-948. ACM, 2017.

[36]. Xiong, Chenyan, Jamie Callan, and Tie-Yan Liu. "Word-entity duet representations for document ranking." In Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 763-772. ACM, 2017.

[37]. Fang C, Mu D, Deng Z, Wu Z. Word-sentence co-ranking for automatic extractive text summarization. Expert Systems with Applications. 2017 Apr 15;72: 189-95.

[38]. Wan, Mengting, and Julian McAuley. "Modeling ambiguity, subjectivity, and diverging viewpoints in opinion question answering systems." Data Mining (ICDM), 2016 IEEE 16th International Conference on. IEEE, 2016.

[39]. McAuley, Julian, and Alex Yang. "Addressing complex and subjective product-related queries with customer reviews." Proceedings of the 25th International Conference on World Wide Web. International World Wide Web Conferences Steering Committee, 2016.

[40]. Etzioni, Oren. "Search needs a shake-up." Nature 476, no. 7358 (2011): 25.

[41]. Sun, Huan, Hao Ma, Wen-tau Yih, Chen-Tse Tsai, Jingjing Liu, and Ming-Wei Chang. "Open domain question answering via semantic enrichment." In Proceedings of the 24th International Conference on World Wide Web, pp. 1045-1055. International World Wide Web Conferences Steering Committee, 2015.

[42]. O'Connor, Brendan, Ramnath Balasubramanyan, Bryan R. Routledge, and Noah A. Smith. "From tweets to polls: Linking text sentiment to public opinion time series." Icwsm 11, no. 122-129 (2010): 1-2.

[43]. Ishikawa, Yoshiharu, and Mikine Hasegawa. "T-scroll: Visualizing trends in a time-series of documents for interactive user exploration." In International Conference on Theory and Practice of Digital Libraries, pp. 235-246. Springer, Berlin, Heidelberg, 2007.

[44]. Ventura, Joao, and Joaquim Ferreira da Silva. "Ranking and extraction of relevant single words in text." In Brain, Vision and ai. InTech, 2008.

[45]. Jurczyk, Pawel, and Eugene Agichtein. "Discovering authorities in question answer communities by using link analysis." In Proceedings of the sixteenth ACM conference on Conference on information and knowledge management, pp. 919-922. ACM, 2007.

[46]. Zhou, Zhi-Min, Man Lan, Zheng-Yu Niu, and Yue Lu. "Exploiting user profile information for answer ranking in cqa." In Proceedings of the 21st international conference on World Wide Web, pp. 767-774. ACM, 2012.

[47]. Jeon, Jiwoon, W. Bruce Croft, Joon Ho Lee, and Soyeon Park. "A framework to predict the quality of answers with non-textual features." In Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 228-235. ACM, 2006.

[48]. Tu, Xudong, Xin-Jing Wang, Dan Feng, and Lei Zhang. "Ranking community answers via analogical reasoning." In Proceedings of the 18th international conference on World wide web, pp. 1227-1228. ACM, 2009.

[49]. Yu, Xiaohui, Ziqiang Yu, Yang Liu, and Huxia Shi. "CI-Rank: collective importance ranking for keyword search in databases." Information Sciences 384 (2017): 1-20.

[50]. Sakoe, Hiroaki, and Seibi Chiba. "Dynamic programming algorithm optimization for spoken word recognition." IEEE transactions on acoustics, speech, and signal processing 26, no. 1 (1978): 43-49.

[51]. Berndt, Donald J., and James Clifford. "Using dynamic time warping to find patterns in time series." In KDD workshop, vol. 10, no. 16, pp. 359-370. 1994.

[52]. Müller, Meinard. Information retrieval for music and motion. Vol. 2. Heidelberg: Springer, 2007.

[53]. Tsinaslanidis, Prodromos, Antonis Alexandridis, Achilleas Zapranis, and Efstratios Livanis. "Dynamic time warping as a similarity measure: applications in finance." (2014).

[54]. Bautista, Miguel Angel, Antonio Hernández-Vela, Victor Ponce, Xavier Perez-Sala, Xavier Baró, Oriol Pujol, Cecilio Angulo, and Sergio Escalera. "Probability-based dynamic time warping for gesture recognition on RGB-D data." In Advances in depth image analysis and applications, pp. 126-135. Springer, Berlin, Heidelberg, 2013.

[55]. Chen, Yanping, Bing Hu, Eamonn Keogh, and Gustavo EAPA Batista. "DTW-D: time series semi-supervised learning from a single example." In Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 383-391. ACM, 2013.

[56]. Russo, J. Edward. "More information is better: A reevaluation of Jacoby, Speller and Kohn." Journal of Consumer Research1, no. 3 (1974): 68-72.

[57]. Blooma, Mohan John, Alton YK Chua, and Dion Hoe-Lian Goh. "A predictive framework for retrieving the best answer." In Proceedings of the 2008 ACM symposium on Applied computing, pp. 1107-1111. ACM, 2008.

[58]. Pande, Vinay, Tanmoy Mukherjee, and Vasudeva Varma. "Summarizing answers for community question answer services." In Language Processing and Knowledge in the Web, pp. 151-161. Springer, Berlin, Heidelberg, 2013.

[59]. Hambleton, Ronald K., and Anil Kanjee. "Increasing the validity of cross-cultural assessments: Use of improved methods for test adaptations." European Journal of Psychological Assessment 11, no. 3 (1995): 147-157.

[60]. Luo, Yi, Xuemin Lin, Wei Wang, and Xiaofang Zhou. "Spark: top-k keyword query in relational databases." In Proceedings of the 2007 ACM SIGMOD international conference on Management of data, pp. 115-126. ACM, 2007.

[61]. Long, Daniel. "Quasi-standard as a linguistic concept." American speech 71, no. 2 (1996): 118-135.

[62]. Ye, Jun. "Multicriteria group decision-making method using vector similarity measures for trapezoidal intuitionistic fuzzy numbers." Group Decision and Negotiation 21, no. 4 (2012): 519-530.

[63]. Wang, Yining, Liwei Wang, Yuanzhi Li, Di He, Wei Chen, and Tie-Yan Liu. "A theoretical analysis of NDCG ranking measures." In Proceedings of the 26th Annual Conference on Learning Theory (COLT 2013), vol. 8. 2013.

[64]. Baltrunas, Linas, Tadas Makcinskas, and Francesco Ricci. "Group recommendations with rank aggregation and collaborative filtering." In Proceedings of the fourth ACM conference on Recommender systems, pp. 119-126. ACM, 2010.

[65]. Jurgens, David, and Ioannis Klapaftis. "Semeval-2013 task 13: Word sense induction for graded and non-graded senses." In Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013), vol. 2, pp. 290-299. 2013.

[66]. Tiun, Sabrina, Rosni Abdullah, and Tang Enya Kong. "Automatic topic identification using ontology hierarchy." In International Conference on Intelligent Text Processing and Computational Linguistics, pp. 444-453. Springer, Berlin, Heidelberg, 2001.

[67]. Moturu, Sai T., and Huan Liu. "Quantifying the trustworthiness of social media content." Distributed and Parallel Databases29, no. 3 (2011): 239-260.

[68]. Lee, Jung-Tae, Min-Chul Yang, and Hae-Chang Rim. "Discovering high-quality threaded discussions in online forums." Journal of Computer Science and Technology 29, no. 3 (2014): 519-531.

[69]. Lee, Jung-Tae, Jangwon Seo, Jiwoon Jeon, and Hae-Chang Rim. "Sentence-based relevance flow analysis for high accuracy retrieval." Journal of the American Society for Information Science and Technology 62, no. 9 (2011): 1666-1675.

[70]. Raiber, Fiana, and Oren Kurland. "Kullback-leibler divergence revisited." Proceedings of the ACM SIGIR International Conference on Theory of Information Retrieval. ACM, 2017.

[71]. Oya, Masaru. "Relation between mechanism of soil removal from fabrics and a parameter derived from probability density functional method for washing force analysis." Textile Research Journal (2018): 0040517518790978.

[72]. Ponti, Moacir, Josef Kittler, Mateus Riva, Teófilo de Campos, and Cemre Zor. "A decision cognizant Kullback–Leibler divergence." Pattern Recognition 61 (2017): 470-478.

[73]. Bušic, Ana, and Sean Meyn. "Action-Constrained Markov Decision Processes With Kullback–Leibler Cost." Proceedings of Machine Learning Research vol 75 (2018): 1-14.

[74]. Ha, Wooseok, Emil Y. Sidky, Rina Foygel Barber, Taly Gilat Schmidt, and Xiaochuan Pan. "Estimating the spectrum in computed tomography via Kullback-Leibler divergence constrained optimization." arXiv preprint arXiv:1805.00162(2018).

[75]. Galas, David J., Gregory Dewey, James Kunert-Graf, and Nikita A. Sakhanenko. "Expansion of the Kullback-Leibler divergence, and a new class of information metrics." Axioms6, no. 2 (2017): 8.

[76]. Delpha, Claude, Demba Diallo, and Abdulrahman Youssef. "Kullback-Leibler Divergence for fault estimation and isolation: Application to Gamma distributed data." Mechanical Systems and Signal Processing 93 (2017): 118-135.

[77]. Li, Zihao, Wenchuan Wu, Boming Zhang, and Xue Tai. "Kullback–Leibler divergence-based distributionally robust optimisation model for heat pump day-ahead

operational schedule to improve PV integration." IET Generation, Transmission & Distribution (2018).

[78]. Maddux, Nathaniel R., Austin L. Daniels, and Theodore W. Randolph. "Microflow imaging analyses reflect mechanisms of aggregate formation: comparing protein particle data sets using the Kullback–Leibler divergence." Journal of pharmaceutical sciences 106, no. 5 (2017): 1239-1248.

[79]. Johnson, Don, and Sinan Sinanovic. "Symmetrizing the kullback-leibler distance." IEEE Transactions on Information Theory (2001).

[80]. Jean Hausser and Korbinian Strimmer. "Package 'entropy'." (2015).

[81]. Mohan, Sunil, Nicolas Fiorini, Sun Kim, and Zhiyong Lu. "A fast deep learning model for textual relevance in biomedical information retrieval." arXiv preprint arXiv:1802.10078 (2018).

[82]. Zhai, Chengxiang, and John Lafferty. "A study of smoothing methods for language models applied to ad hoc information retrieval." In ACM SIGIR Forum, vol. 51, no. 2, pp. 268-276. ACM, 2017.

[83]. Turtle, Howard, and W. Bruce Croft. "Inference networks for document retrieval." In ACM SIGIR Forum, vol. 51, no. 2, pp. 124-147. ACM, 2017.

[84]. Berger, Adam, and John Lafferty. "Information retrieval as statistical translation." In ACM SIGIR Forum, vol. 51, no. 2, pp. 219-226. ACM, 2017.

[85]. Yoon, Taewon, Sung-Hyon Myaeng, Hyun-Wook Woo, Seung-Wook Lee, and Sang-Bum Kim. "On Temporally Sensitive Word Embeddings for News Information Retrieval." NewsIR@ ECIR 2079 (2018): 51-56.

[86]. Xu, Jinxi, and W. Bruce Croft. "Quary expansion using local and global document analysis." In Acm sigir forum, vol. 51, no. 2, pp. 168-175. ACM, 2017.

[87]. Agarwal, Sumeet, Shantanu Godbole, Diwakar Punjani, and Shourya Roy. "How much noise is too much: A study in automatic text classification." In Data Mining, 2007. ICDM 2007. Seventh IEEE International Conference on, pp. 3-12. IEEE, 2007.

[88]. Apostolova, Emilia, and R. Andrew Kreek. "Training and Prediction Data Discrepancies: Challenges of Text Classification with Noisy, Historical Data." arXiv preprint arXiv:1809.04019 (2018).

[89]. Nguyen, Hoang, and Jon Patrick. "Text Mining in Clinical Domain: Dealing with Noise." In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 549-558. ACM, 2016.

[90]. Li, Ximing, Yue Wang, Ang Zhang, Changchun Li, Jinjin Chi, and Jihong Ouyang. "Filtering out the noise in short text topic modeling." Information Sciences 456 (2018): 83-96.

[91]. Xiang, Lingyun, Jiaohua Qin, Xiao Yang, and Qichao Tang. "An Adaptive Steganographic Method Using Additive Noise." JCP 11, no. 3 (2016): 207-215.

[92]. Patel, Charmi, and Hiteishi Diwanji. "A Research on Web Content Extraction and Noise Reduction through Text Density Using Malicious URL Pattern Detection." (2016).

[93]. Shannon, C.E., 1948. A mathematical theory of communication. Bell system technical journal, 27(3), pp.379-423.

[94]. Zhao, Yang, and Fangai Liu. "A clustering algorithm based on maximum entropy principle." In Journal of Physics: Conference Series, vol. 887, no. 1, p. 012064. IOP Publishing, 2017.

[95]. Btoush, Mohammad Hjouj, and Ziad E. Dawahdeh. "A Complexity Analysis and Entropy for Different Data Compression Algorithms on Text Files." Journal of Computer and Communications 6, no. 01 (2017): 301.

[96]. Abualigah, Laith Mohammad, Ahamad Tajudin Khader, Mohammed Azmi Al-Betar, and Mohammed A. Awadallah. "A krill herd algorithm for efficient text documents clustering." In 2016 IEEE symposium on computer applications & industrial electronics (ISCAIE), pp. 67-72. IEEE, 2016.

[97]. Abbas, Ali, Andrea H Cadenbach, and Ehsan Salimi. "A Kullback–Leibler View of Maximum Entropy and Maximum Log-Probability Methods." Entropy 19, no. 5 (2017): 232.

[98]. He, Yonghuan, Hongwei Guo, Maozhu Jin, and Peiyu Ren. "A linguistic entropy weight method and its application in linguistic multi-attribute group decision making." Nonlinear Dynamics84, no. 1 (2016): 399-404.

[99]. Revanasiddappa, M. B., and B. S. Harish. "A New Feature Selection Method based on Intuitionistic Fuzzy Entropy to Categorize Text Documents." International Journal of Interactive Multimedia & Artificial Intelligence 5, no. 3 (2018).

[100]. Zhang, Hui, Kaihu Hou, and Zhou Zhou. "A Weighted KNN Algorithm Based on Entropy Method." In Intelligent Computing and Internet of Things, pp. 443-451. Springer, Singapore, 2018.

[101]. Zou, Baoping. "Accurate Text Classification via Maximum Entropy Model." In International Conference on Collaborative Computing: Networking, Applications and Worksharing, pp. 569-576. Springer, Cham, 2016.

[102]. Zhang, Ye, Matthew Lease, and Byron C. Wallace. "Active discriminative text representation learning." In Thirty-First AAAI Conference on Artificial Intelligence. 2017.

[103]. Romero, Verónica, Joan Andreu Sánchez, and Alejandre H. Toselli. "Active Learning in Handwritten Text Recognition using the Derivational Entropy." In 2018 16th International Conference on Frontiers in Handwriting Recognition (ICFHR), pp. 291-296. IEEE, 2018.

[104]. Zheng, Wenbo, Shaocong Mo, Pengfei Duan, and Xiaotian Jin. "An improved pagerank algorithm based on fuzzy C-means clustering and information entropy." In 2017 3rd IEEE International Conference on Control Science and Systems Engineering (ICCSSE), pp. 615-618. IEEE, 2017.

[105]. Namazi, Hamidreza, Amin Akrami, Reza Haghighi, Ali Delaviz, and Vladimir V. Kulish. "Analysis of the Influence of Element's Entropy on the Bulk Metallic Glass (BMG) Entropy, Complexity, and Strength." Metallurgical and Materials Transactions A 48, no. 2 (2017): 780-788.

[106]. Bierig, Claudio, and Alexey Chernov. "Approximation of probability density functions by the Multilevel Monte Carlo Maximum Entropy method." Journal of Computational Physics314 (2016): 661-681.

[107]. Laleye, Fréjus AA, Eugène C. Ezin, and Cina Motamed. "Automatic text-independent syllable segmentation using singularity exponents and rényi entropy." Journal of Signal Processing Systems 88, no. 3 (2017): 439-451.

[108]. Kan, Jeff WT, and John S. Gero. "Characterizing innovative processes in design spaces through measuring the information entropy of empirical data from protocol studies." AI EDAM 32, no. 1 (2018): 32-43.

[109]. Dredze, Mark, Hanna M. Wallach, Danny Puller, and Fernando Pereira. "Generating summary keywords for emails using topics." In Proceedings of the 13th international conference on Intelligent user interfaces, pp. 199-206. ACM, 2008.

[110]. Wartena, Christian, Wout Slakhorst, and Martin Wibbels. "Selecting keywords for content based recommendation." In Proceedings of the 19th ACM international conference on Information and knowledge management, pp. 1533-1536. ACM, 2010.

[111]. Kommers, Jefferson M., David Freed, and Damien Paul Kennedy. "Information retrieval from a collection of information objects tagged with hierarchical keywords." U.S. Patent 7,028,024, issued April 11, 2006.

[112]. Bennett, Casey C., and Kris Hauser. "Artificial intelligence framework for simulating clinical decision-making: A Markov decision process approach." Artificial intelligence in medicine57, no. 1 (2013): 9-19.

[113]. Tiomkin, Stas, and Naftali Tishby. "A Unified Bellman Equation for Causal Information and Value in Markov Decision Processes." arXiv preprint arXiv:1703.01585 (2017).

[114]. Pollock, Felix A., César Rodríguez-Rosario, Thomas Frauenheim, Mauro Paternostro, and Kavan Modi. "Operational Markov condition for quantum processes." Physical review letters 120, no. 4 (2018): 040405.

[115]. Kang, Mangi, Jaelim Ahn, and Kichun Lee. "Opinion mining using ensemble text hidden Markov models for text classification." Expert Systems with Applications 94 (2018): 218-227.

[116]. George, Mishel, Saber Jafarpour, and Francesco Bullo. "Markov chains with maximum entropy for robotic surveillance." IEEE Transactions on Automatic Control (2018).

[117]. Wald, A., and Wolfowitz, J. (1940), "On a Test Whether Two Samples are From the Same Population," Annials of Mathematical Statistics, 11, 147-162.

[118]. Leauhatong, Thurdsak, Kazuhiko Hamamoto, Kiyoaki Atsuta, and Shozo Kondo. "A New Content-Based Image Retrieval Using the Multidimensional Generalization of Wald-Wolfowitz Runs Test." IEEJ Transactions on Electronics, Information and Systems 129, no. 1 (2009): 94-102.

[119]. Magel, Rhonda C., and Sasmito H. Wibowo. "Comparing the powers of the Wald-Wolfowitz and Kolmogorov-Smirnov tests." Biometrical Journal 39, no. 6 (1997): 665-675.

[120]. Mohanta, Partha Pratim, Sanjoy Kumar Saha, and Bhabatosh Chanda. "Detection of representative frames of a shot using multivariate wald-wolfowitz test." In 2008 19th International Conference on Pattern Recognition, pp. 1-4. IEEE, 2008.

[121]. Song, Shengyuan, Qing Wang, Jianping Chen, Yanyan Li, Qi Zhang, and Chen Cao. "A multivariate method for identifying structural domain boundaries in a rock mass." Bulletin of Engineering Geology and the Environment 74, no. 4 (2015): 1407-1418.

[122]. George, Anjith, and Aurobinda Routray. "A score level fusion method for eye movement biometrics." Pattern Recognition Letters 82 (2016): 207-215.

[123]. Kovačević, Strahinja Z., Aleksandra N. Tepić, Lidija R. Jevrić, Sanja O. Podunavac-Kuzmanović, Senka S. Vidović, Zdravko M. Šumić, and Žarko M. Ilin. "Chemometric guidelines for selection of cultivation conditions influencing the antioxidant potential of beetroot extracts." Computers and Electronics in Agriculture 118 (2015): 332-339.

[124]. Chen, Chao W., Dennis Hsieh, Fung-Chang Sung, and Shan P. Tsai. "Feasibility of using urinary TDGA as a biomarker for VCM exposures." Regulatory Toxicology and Pharmacology (2018).

[125]. Song, Shengyuan, Fengyue Sun, Wen Zhang, Jianping Chen, Peihua Xu, Cencen Niu, Chen Cao, and Jiewei Zhan. "Identification of structural domains by considering multiple discontinuity characteristics: a case study of the Songta Dam." Bulletin of Engineering Geology and the Environment 77, no. 4 (2018): 1589-1598.

[126]. Linkowska, Katarzyna, Arkadiusz Jawień, Andrzej Marszałek, Boris A. Malyarchuk, Katarzyna Tońska, Ewa Bartnik, Katarzyna Skonieczna, and Tomasz Grzybowski. "Mitochondrial DNA polymerase γ mutations and their implications in mtDNA alterations in colorectal cancer." Annals of human genetics 79, no. 5 (2015): 320

[127]. Chapelle, Olivier, Bernhard Scholkopf, and Alexander Zien. "Semi-supervised learning (chapelle, o. et al., eds.; 2006)[book reviews]." IEEE Transactions on Neural Networks 20, no. 3 (2009): 542-542.

[128]. Laskov, Pavel, Patrick Düssel, Christin Schäfer, and Konrad Rieck. "Learning intrusion detection: supervised or unsupervised?." In International Conference on Image Analysis and Processing, pp. 50-57. Springer, Berlin, Heidelberg, 2005.

[129]. Zhu, Xiaojin, and Andrew B. Goldberg. "Introduction to semi-supervised learning." Synthesis lectures on artificial intelligence and machine learning 3, no. 1 (2009): 1-130.

[130]. Sheikhpour, Razieh, Mehdi Agha Sarram, Sajjad Gharaghani, and Mohammad Ali Zare Chahooki. "A Survey on semi-supervised feature selection methods." Pattern Recognition 64 (2017): 141-158.

[131]. Caruana, Rich, and Alexandru Niculescu-Mizil. "An empirical comparison of supervised learning algorithms." In Proceedings of the 23rd international conference on Machine learning, pp. 161-168. ACM, 2006.

[132]. Memeti, Suejb, and Sabri Pllana. "PAPA: A parallel programming assistant powered by IBM Watson cognitive computing technology." Journal of computational science 26 (2018): 275-284.

[133]. Mayeesha, Tahsin, Zareen Tasneem, Jasmine Jones, and Nova Ahmed. "Applying Text Mining to Protest Stories as Voice against Media Censorship." arXiv preprint arXiv:1812.11430 (2018).

Appendices

- **R Codes of DTW distance:**

```
for(i in 1:"Total File number"){
  d <- c()
  s <- paste("the Zero/One file address",i,sep="")
  s <- paste(s,".txt",sep="")
  originaldat <- readLines(s)
  dat <- unlist(strsplit(originaldat, ",", fixed = TRUE))
  thelen = length(dat)
  for(j in 1:thelen){d <- c(d,as.numeric(unlist(dat[j])))}
  alignment <- dtw(the_Ideal_answer,d,keep=TRUE)
}
```

- **R Codes of $g(Q)$:**

```
Q_d <- c()
Q_originaldat <- readLines("the Address of Global Probability")
Q_dat <- unlist(strsplit(Q_originaldat, ",", fixed = TRUE))
Q_thelen = length(Q_dat)
for(j in 1:Q_thelen){Q_d <- c(Q_d,as.numeric(unlist(Q_dat[j])))}
q.f <- function(lambda){sum(exp(-1-(lambda/Q_d)))-1}
uniroot(q.f, c(-10000,10000))$root
```

Curriculum Vitae

Name:	GuanDong Zhang
Post-secondary Education and Degrees:	Hangzhou Dianzi University, HangZhou, Zhejiang, China 1998-2002 B.A. Chengdu University of Technology, Chengdu, SiChuan, China 2003-2006 M.A.
Related Work Experience	Research Assistant Centre for Oral History and Digital Storytelling (COHDS), Concordia University 2017.5-2017.8

Publications:

Zhang, GuanDong, Hao Yu, and Lu Xiao. "CEW-DTW: A new time series model for text mining." In 2018 International Conference on Information and Communications Technology (ICOIACT), pp. 158-162. IEEE, 2018.