International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Impact Factor (2012): 3.358

A Survey on a New Approach for Improving Efficiency and Accuracy in String Transformation

Swapnali S. Maske[1], Prashant Jawalkar [2]

[1]ME Student, Department of computer Engineering,
JSPMs B.S.I.O.T.R, Wagholi, Pune University, Pune, Maharashtra, India

[2]Assistant Professor, Engineering, Department of computer Engineering,
JSPMs B.S.I.O.T.R, Wagholi, Pune University, Pune, Maharashtra, India

Abstract: ***The method of string transformation includes different tasks such as record matching in database, spelling error correction, reformulating query & mining the synonyms. The likelihood of transformation can represent similarity, relevance & association between two strings in specific application but from the viewpoint of enhancing both accuracy and efficiency string transformation this method involves correction of spelling errors in queries as well as reformulation of queries in web search. Corresponding spelling errors in queries usually consist of two steps: candidate generation & candidate selection. Candidate generation is concerned with single word. This method is using top k pruning algorithm and efficient dictionary matching algorithm. The method works on two problems: spelling error correction of queries and reformulation of queries in web search. The difference between the two problems is that string transformation is performed at a character level in the former task and word level in latter task.***

Keywords: Minimum Description Length (MDL), Argumentive and Alternative Communication (AAC), Conditional Random Field Query Refinement (CRF-QR), Query Reformulation (QE).

1. Introduction

The necessity of Dictionary is to finding meaning for the specific word or vocabulary. When the word in weak entity means correcting each word in sentences could not be possible. Input query do not match well and the file will not be graded high. Efficiency is an important factor taken into deliberation in our process. Detachment does not take setting data into consideration. At the end count for the string pair only limited leverages. This is only the reason for the input variances. Introduces many correction and transformation technique did not generate better candidates. However the development would not significant at any module which was associating the number of input process only having small execution. Typical process execution leads the time consumption for the full implementation. Till the weak wordings on any sentence not yet give the full correction dynamically. Accuracy for pattern matching system need to give more outcomes.

Word prediction uses language modeling, where within a set vocabulary the words are most likely to occur are calculated. Along with language modeling, basic word prediction on AAC devices is often coupled with a regency model, where words that are used more frequently by the AAC user are more likely to be predicted. Word prediction software often also allows the user to enter their own words into the word prediction dictionaries either directly, or by "learning" words that have been written. From our project we are referring the input query process with methodological references. According to the Minimum description length compression can be made to explore the input data. At the end naïve based algorithm helps to find the correctness of the string. We are considering the term pattern for the original word. Taking the input string named as text for comparing both words. We are under taking this kind of process in the entire paper.

2. Literature Survey

2.1 Online Spelling Correction for Query Completion [14]

This model proposes a transform based transformation model that is capable of capturing users spelling behavior. Also estimate the transformation model using clicks on search engine recourse links, which represent user confirmed query misspellings. The A* search algorithm is configured to deal with partial queries, so that online search is possible. In this paper, they model search queries with a generative model, where the intended query is transformed through a noisy channel into a potentially misspelled query. The distribution from which the target query is selected is estimated from the search engine query log based on frequency. Thus, they are more likely to suggest more popular queries. For the noisy channel, this describes the distribution of spelling errors.

Main problem with this heuristic function is that it does not penalize the untransformed part of the input query. Therefore, this can design a better heuristic by taking into consideration the upper bound of the transformation probability.

Advantages

1. Making use of both absolute pruning and relative pruning method to improve the search efficiency and accuracy.
2. User can add or drop letters unintentionally by using the process.
3. Suggesting algorithm is not only accurate, but also efficient here

.

Licensed Under Creative Commons Attribution CC BY

Disadvantages

1. Inside the process need suggestions incurs a cost, as users spend more time looking at them instead of completing their task.
2. Irrelevant suggestions risks annoying users to terminate the process.
3. Online correction has many merits that cannot be achieved by offline correction.
4. The algorithm is not sufficiently robust and scalable for online spelling correction for query completion.

2.2 A Unified and Discriminative Model for Query Refinement [8]

This Model describes our new CRF model for performing query refinement, called CRF-QR. The model is unique in that it predicts a sequence of refined query words as well as corresponding operations given a sequence of query words. And show that employing a unified and discriminative model in query refinement is effective. They propose exploiting a unified and discriminative model in query refinement, specifically (1) conducting various query refinement tasks in a unified framework, and (2) employing a special CRF model called CRF-QR to accomplish the tasks. One advantage of employing CRF-QR is that the accuracy of query refinement can be enhanced. This is because the tasks of query refinement are often mutually dependent, and need to be addressed at the same time. Lexicon-based feature representing whether a query word or a refined query word is in a lexicon or a stop word list.

Position-based feature representing whether a query word is at the beginning, middle, or end of the query. Word-based feature representing whether a query word consists of digit, alphabet, or a mix of the two, and whether the length of a query word is in a certain range. Corpus-based feature representing whether the frequency of a query word or a query word in the corpus exceeds a certain threshold. Query-based feature representing whether the query is a single word query or multi-word query.

Advantages

1. Progress will get guaranteed that the global optimal solution will be found because the log-likelihood function is convex.
2. Here heuristics used to reduce the number of possible sequences inside the process.

Disadvantages

1. Efficiency of this model performance is comparatively low than our string transformation system
2. The use of the basic model is not enough for the correction and stemming process.

Word stemming it is not easy to make a refinement judgment, be-cause the effectiveness of a refinement also depends on the contents of document collection.

2.3 Space-Constrained Gram-Based Indexing For Efficient Approximate String Search [9]

In this paper they study how to reduce the size of such index structures, while still maintaining a high query performance. The setting of approximate string search is unique in that a candidate result needs to occur at least a certain number of times among all the inverted lists, and not necessarily on all the inverted lists. The first approach is based on the idea of discarding some of the lists. They study several technical challenges that arise naturally in this approach. One issue is how to compute a new lower bound on the number of common grams (whose lists are not discarded) shared by two similar strings, the formula of which becomes technically interesting. These models partition an inverted list into fixed-size segments and compress each segment with a word-aligned integer coding scheme. Also studied how to adopt existing inverted-list compression techniques to achieve the goal, and proposed two novel methods for achieving the goal.

The trie based pruning is not included in this model; hence performance of this model is not so efficient.

Advantages

1. They have used two steps to discovering candidate gram pairs and selecting some of them to combine.
2. It has construct to be more effective because data sets used different reduction ratios for equal the limitation of technique.

Disadvantages

1. Estimation is not 100% accurate, and an inaccurate result could greatly affect the accuracy of the estimated post-processing time. This will affect the quality of the selected non-whole lists.
2. This estimation may need to be done repeatedly when choosing lists to discard, and therefore needs to be very efficient but it has failed to do that.

2.4 Exploring Distributional Similarity Based Models for Query Spelling Correction [6]

Mu Li et.al concentrate on the problem of learning improved query spelling correction model by integrating distributional similarity information automatically derived from query logs. The key contribution of our work is identifying that we can successfully use the evidence of distributional similarity to achieve better spelling correction accuracy.

We present two methods that are able to take advantage of distributional similarity information. The First method extends a string edit based error Model with confusion probabilities within a generative source channel model. The second Method explores the effectiveness of our approach within a discriminative maximum entropy model framework by integrating distributional similarity based features.

Advantages

1. We used the standard Viterbi algorithm to search for the best output of source channel model.
2. The distributional similarity known to achieve better spelling correction accuracy.

Licensed Under Creative Commons Attribution CC BY

Disadvantages

1. The paper reports that un-weighted edit distance will cause the overall accuracy of their speller's output.
2. Probability renormalization and smoothing problem has been thrown on to the process.

2.5 A Discriminative Candidate Generator for String Transformations [16]

This paper addresses these challenges by exploring the discriminative training of candidate generators. More specifically, we build a binary classifier that, when given a source strings, decides whether a candidate t should be included in the candidate set o not. This approach appears straightforward, but it must resolve two practical issues. First, the task of the classifier is not only to make a binary decision for the two strings s and t, but also to enumerate a set of positive strings for the string s.

Another issue arises when we prepare a training set. A discriminative model requires a training set in which each instance (pair of strings) is annotated with a positive or negative label. They design features that express transformations from a source string s to its destination string t. And also present an algorithm that utilizes the feature weights to enumerate candidates of destination strings efficiently.

Advantages

1. Inserting the vocabulary into a suffix array, this is used to locate every occurrence on process easily.
2. Trained CST's lemmatiser for each dataset to obtain flex patterns that are used for normalizing inflections exactly.

So they can proceeds the cross word evaluation

Disadvantages

1. Generation algorithm of substitution rules had produced inappropriate rules that transform a string incorrectly.
2. Finding distance or similarity metrics did not specifically derive destination strings to which the classifier is likely to assign positive labels.
3. We could not use the efficient algorithm as a candidate generator with all required features. Due to this will lead to loss some words.
4. Some process is not suited for a candidate generator because the processes of string transformations are intractable in their discriminative models.

3. Conclusion

This paper states the difficulty of spelling correction for search queries by adopting a generative model for query correction. To efficiently retrieve the query corrections with the highest probability, unique model, machine learning process and string matching algorithm. Two specific applications are addressed with this survey, namely spelling changing of input string and naïve based verification. We can propose a numerical learning approach to find the error on specific sequences on input query. Finally, we can propose a naïve based matching algorithm for best correctness which is more accurate and efficient.

References

[1] Arasu, S. Chaudhuri, and R. Kaushik, "Learning string transformations from examples," Proc. VLDB Endow., vol. 2, pp. 514– 525, August 2009.

[2] S. Tejada, C. A. Knoblock, and S. Minton, "Learning domain independent string transformation weights for high accuracy object identification," in Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining, ser. KDD '02. New York, NY, USA: ACM, 2002, pp. 350–359.

[3] M. Hadjieleftheriou and C. Li, "Efficient approximate search on string collections," Proc. VLDB Endow., vol. 2, pp. 1660–1661, August 2009.

[4] C. Li, B. Wang, and X. Yang, "Vgram: improving performance of approximate queries on string collections using variable-length grams," in Proceedings of the 33rd international conference on Very large data bases, ser. VLDB '07. VLDB Endowment, 2007, pp. 303–314.

[5] X. Yang, B. Wang, and C. Li, "Cost-based variable-length-gram selection for string collections to support approximate queries efficiently," in Proceedings of the 2008 ACM SIGMOD international conference on Management of data, ser. SIGMOD '08. New York, NY, USA: ACM, 2008, pp. 353–364.

[6] M. Li, Y. Zhang, M. Zhu, and M. Zhou, "Exploring distributional similarity based models for query spelling correction," in Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics, ser. ACL '06. Morristown, NJ, USA: Association for Computational Linguistics, 2006, pp. 1025–1032.

[7] A. R. Golding and D. Roth, "A winnow-based approach to context-sensitive spelling correction," Mach. Learn., vol. 34, pp.107–130, February 1999.

[8] J. Guo, G. Xu, H. Li, and X. Cheng, " A unified and discriminative model for query refinement," in Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval, ser. SIGIR '08. New York, NY, USA: ACM, 2008, pp. 379–386.

[9] A. Behm, S. Ji, C. Li, and J. Lu, "Space-constrained gram-based indexing for efficient approximate string search," in Proceedings of the 2009 IEEE International Conference on Data Engineering, ser.ICDE '09. Washington, DC, USA: IEEE Computer Society, 2009,pp. 604–615.

[10] E. Brill and R. C. Moore, "An improved error model for noisy channel spelling correction," in Proceedings of the 38th Annual Meeting on Association for Computational Linguistics, ser. ACL '00.Morristown, NJ, USA: Association for Computational Linguistics,2000, pp. 286–293.

[11] M. Dreyer, J. R. Smith, and J. Eisner, "Latent-variable modeling of string transductions with finite-state methods," in Proceedings of the Conference on Empirical Methods in Natural Language Processing, ser. EMNLP '08. Stroudsburg, PA, USA: Association for Computational Linguistics, 2008, pp. 1080–1089.

Licensed Under Creative Commons Attribution CC BY

[12] E. S. Ristad and P. N. Yianilos, "Learning string-edit distance," IEEE Trans. Pattern Anal. Mach. Intell., vol. 20, pp. 522–532, May1998.
[13] J. Oncina and M. Sebban, "Learning unbiased stochastic edit distance in the form of a memory less finite-state transducer," in InWorkshop on Grammatical Inference Applications: Successes and Future Challenges, 2005.
[14] H. Duan and B.-J. P. Hsu, "Online spelling correction for query completion," in Proceedings of the 20th international conference on World Wide Web, ser. WWW '11. New York, NY, USA: ACM, 2011, pp. 117–126.
[15] Q. Chen, M. Li, and M. Zhou, "Improving query spelling correction using web search results," in Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, ser. EMNLP '07, 2007,pp. 181–189.
[16] N. Okazaki, Y. Tsuruoka, S. Ananiadou, and J. Tsujii, "A discriminative candidate generator for string transformations," in Proceedings of the Conference on Empirical Methods in Natural Language Processing, ser. EMNLP '08. Morristown, NJ, USA: Association for Computational Linguistics, 2008, pp. 447–456.

Author Profile



Swapnali S. Maske is ME Student, Department Of Computer Engineering, JSPMS B.S.I.O.T.R., Wagholi, Pune University, Pune, Maharashtra, India



Prashant Jawalkar is Assistant Professor, Engineering, Department of computer Engineering, JSPMs B.S.I.O.T.R, Wagholi, Pune university, Pune, Maharashtra, India

Licensed Under Creative Commons Attribution CC BY