An Ensemble Method for Spelling Correction in Consumer Health Questions

Halil Kilicoglu, PhD, Marcelo Fiszman, MD, PhD, Kirk Roberts, PhD,
Dina Demner-Fushman, MD, PhD
Lister Hill National Center for Biomedical Communications
U.S. National Library of Medicine
Bethesda, MD

Abstract

Orthographic and grammatical errors are a common feature of informal texts written by lay people. Health-related questions asked by consumers are a case in point. Automatic interpretation of consumer health questions is hampered by such errors. In this paper, we propose a method that combines techniques based on edit distance and frequency counts with a contextual similarity-based method for detecting and correcting orthographic errors, including misspellings, word breaks, and punctuation errors. We evaluate our method on a set of spell-corrected questions extracted from the NLM collection of consumer health questions. Our method achieves a F_1 score of 0.61, compared to an informed baseline of 0.29, achieved using ESpell, a spelling correction system developed for biomedical queries. Our results show that orthographic similarity is most relevant in spelling error correction in consumer health questions and that frequency and contextual information are complementary to orthographic features.

1 Introduction

Orthographic errors are pervasive in informal writing. The questions that consumers ask about their or someone else's health often contain many misspellings[1]. Misspellings may not pose a significant cognitive burden for a human reader, but they can severely limit the effectiveness of an automated system. At the National Library of Medicine (NLM), we have been building a system to assist customer service staff in answering health questions received from consumers. In 2014, we received more than 40K questions, approximately 15% of which sought health-related information. Our system currently uses a combination of rule-based and supervised machine learning techniques for question understanding. More specifically, we extract *focus* (generally a disease)[2] and *question type* (e.g., treatment, prognosis)[3] from the question and construct a semantic frame[4], which is then converted to a search engine query. These techniques assume well-written questions; thus, orthographic errors significantly hinder their performance. For instance, consider the following question:

(1) *My mom is 82 years old suffering from anixity and depression for the last 10 years was dianosed early on set deminita 3 years ago. Do yall have a office in Greensboro NC? Can you recommend someone. she has seretona syndrome and nonething helps her.*

Four disorders are mentioned in the question, three of which are misspelled (*anixity* for *anxiety*, *deminita* for *dementia*, and *seretona syndrome* for *serotonin syndrome*). Another, perhaps less central but potentially important, misspelling is *dianosed* for *diagnosed*. On the other hand, the misspelling of the colloquial *yall* (*y'all*) and *nonething* (*nothing*) may be less significant for interpretation of this question. In contrast to such *non-word* spelling errors, the spelling of *onset* as *on set* constitutes a *real word* spelling error as well as a *word break* error, since both *on* and *set* are valid words. Real word spelling errors can be even more problematic, as in the question *"What can I do to lesson the severity of the adema?"*, where the underlined words, both crucial for understanding the question, are misspelled. Without detecting and correcting these

types of errors, there is little hope of extracting the information needed from this question to answer it automatically.

Punctuation errors can also have an impact on question understanding. Omitting sentence-ending periods or space after punctuation, for example, are likely to cause syntactic parsing errors and, consequently, errors in extracted information. A request, particularly rich in such errors, is given below. Unable to identify sentence-ending periods, Stanford syntactic parser[5] has difficulty parsing this request.

(2) *chromosome 3 found in the bloods between the father and son,,,would this mean that my son,s blood is not the same as mine,,i was told it was all about learning problems,,,but i am woried that theres more involved,,can you send me a chart or some thing describeing ch 3 is all about and too what area of the body its being tested on espicaly a 9 year old child ,,thanks*

In this paper, we present a method for detecting and correcting orthographical errors in consumer health questions. At this time, we do not attempt to correct grammatical errors. The method follows a pipeline architecture and consists of several modules: a) a pre-processing module that specifically focuses on errors involving punctuation and numbers, b) a misspelling detection module that relies on an expanded English dictionary, c) a spelling suggestion module that uses phonetic and orthographic distance, and d) a re-ranking module that uses a linear-weighted ensemble of several algorithms that score these suggestions. The scoring algorithms rely on orthographic, phonetic and contextual similarity as well as corpus frequency. We calculate contextual similarity using word embeddings[6], a recent natural language processing technique based on the distributional hypothesis. To evaluate our method, we developed a dataset that consists of 472 consumer health questions received by NLM, which we manually corrected for spelling errors. We compared the performance of our method against a strong baseline that relies on the ESpell algorithm[7], developed for PubMed query correction. Our results demonstrate the varying degrees of difficulty in correcting specific subtypes of spelling errors: while spelling errors in question elements most salient for question understanding (e.g., disorders) can be corrected to a large extent, real word spelling errors remain challenging.

2 Related Work

The approaches to correcting non-word spelling errors are reviewed by Kukich[8] and Mitton[9]. These approaches often rely on the availability of a comprehensive spelling dictionary and use edit distance[10,11] and phonetic similarity between the misspelled word and the candidate suggestion. More recent approaches incorporate word frequency data collected from large corpora[9,12] as well as contextual information[12]. Flor and Futagi[12] use word frequency data as well as orthographic and phonetic similarity, and re-rank the candidate suggestions using context, achieving very good correction results in correcting misspellings in non-native student essays. They report local error density (misspelling of adjacent words) and competition among inflectional variants as the main sources of their errors.

Real word spelling errors (also called *malapropisms*) are more challenging, because it is impossible to detect such errors in isolation. Thus, they often go undetected by spell checkers. Methods for real word error correction have used semantic information from lexical resources or relied on machine learning techniques and language models, all essentially taking some form of contextual information into account. The first approach is based on the hypothesis that the more distant a word is semantically from the other words in a text, the more likely it is a real word error[13]. WordNet[14] has often been used for calculating word distance. Machine learning techniques often use pre-defined confusion sets (e.g., {*their,there*},{*principle,principal*}) and attempt to learn the typical context for each member using features from adjacent words[15]. Language model-based approaches rely on n-gram probabilities[16,17] generally drawn from web-scale data, such as

Google Web 1T dataset[1]. Syntactic and distributional information has also been used for this task[18]. While syntactic information based on parse features proved useful, the contribution of distributional information based on word cooccurrence was found to be limited.

In the biomedical domain, Crowell *et al.*[19] use a frequency-based technique to improve on existing spelling correction tools, ASpell and GSpell, for non-word spelling correction of consumer health information queries. They use MedlinePlus queries to generate word-frequency statistics. Their results show the significant contribution of frequency-based re-ranking for health information retrieval. Wilbur *et al.*[7] focus on spelling correction in the PubMed search engine. Their approach is based on the noisy channel model and makes use of statistics harvested from the user logs to estimate the probabilities of different types of edits that lead to misspellings. Word frequency counts in the PubMed database are used for computing prior probabilities. They apply different constraints based on the edit distance between the misspelling and the candidate suggestion. Ruch *et al.*[20] use the syntactic and semantic context to improve correction accuracy in clinical records. A simple, edit distance-based correction strategy is augmented with ranking via morpho-syntactic and word sense disambiguation. A named-entity extractor is used to avoid correcting physician and patient names. Patrick *et al.*[21] use a combination of a rule-based suggestion generation system and context-based and frequency-based ranking algorithm for spelling correction in clinical notes. Context-based ranking uses a trigram language model with the 3-word window around the misspelling. Each ranking method achieves the best result on one of the two test corpora. They also report inflectional variants as a challenge in spelling correction.

Word embeddings (also known as *context-predicting models* or *neural language models*)[6,22] are a recent development in distributional semantics research, where the paradigm is to use vectors to represent the contexts that a word appears in and to apply vector algebra techniques to measure similarity between word vectors. In contrast to the more traditional distributional methods that rely on word counts, context-predicting models cast the problem as a supervised learning task and try to maximize the probability of the contexts that the word is observed in the corpus. The technique has been applied successfully to various word similarity tasks (e.g., semantic relatedness, analogy).

3 Methods

We begin this section by discussing our spelling correction dataset. We then explain the modules that comprise our spelling error detection and correction pipeline.

3.1 Dataset

We manually annotated and corrected 472 health-related questions posed to NLM by consumers for orthographic and punctuation errors. We considered the following types of errors in annotation:

- NON-WORD: the misspelled word does not appear in the dictionary (e.g., *physians* for *physicians*)
- REAL-WORD: the misspelled word appears in the dictionary (e.g., *leave* for *live*)
- PUNCTUATION: a spelling error caused by absence of punctuation or a spurious punctuation (e.g., *Ive* for *I've*)

[1] http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC2006T13

- TO-MERGE: a word break error, where a spurious space is introduced to a word (e.g., *dur ing* for *during*)
- TO-SPLIT: a word break error, where two adjacent words run together (e.g., *knowabout* for *know about*)

In addition, we annotated whether the error occurred in a *focus* element of the question (IMPORTANT-FOCUS) or whether it was important for extracting the semantic frame (IMPORTANT-FRAME). All errors annotated as IMPORTANT-FOCUS can also be considered IMPORTANT-FRAME, since one major element of a frame is the disease in focus. A misspelled word may be annotated with multiple types of errors. For example, *onset* misspelled as *on set* is a case of both TO-MERGE and REAL-WORD errors. In Example 1, *serotonin* misspelled as *seretona* is annotated as IMPORTANT-FOCUS, as well, since *serotonin syndrome* is the focal disease in the question.

The annotation was carried out by one of the authors of this paper (MF). 1008 spelling errors were annotated on a total of 1075 tokens. Of these 1008 errors, 39 were labeled as IMPORTANT-FRAME and 96 as IMPORTANT-FOCUS. The distribution of error types annotated in the dataset is given in Table 1.

Table 1: Distribution of spelling error types

Spelling error type	*Frequency*
NON-WORD	436
REAL-WORD	154
PUNCTUATION	58
TO-MERGE	45
TO-SPLIT	315
TOTAL	1008

3.2 Pipeline

The spelling error detection and correction pipeline, illustrated in Figure 1, consists of four modules. The *preprocessor* uses simple heuristics to correct punctuation and splitting errors that occur frequently and are difficult to correct using the methods applied to non-word or real word errors. For example, consumer health questions contain many contractions (e.g., *i'm*, *there's*) and they are often spelled without the apostrophe. These are usually short words and most systems do not attempt to correct them, since the number of candidates based on edit distance and phonetic similarity is generally high. Therefore, we simply substitute such errors in preprocessing. For this purpose, we used Wikipedia's list of English contractions[2]. We also correct some informal expressions, such as *plz* for *please* and *u* for *you*. Additionally, we attempt to correct punctuation errors that could cause downstream tokenization or parsing errors, such as a punctuation without a space following it. In the example below, *male.reading* is taken as a single token, which would lead the system to miss the important information about the patient (*male*).

(3) *hi im a support worker for a 46 yr old autistic male.reading your info provided about wolfram syndrome ,he matches alot of the symptoms.*

[2]http://en.wikipedia.org/wiki/Wikipedia:List_of_English_contractions

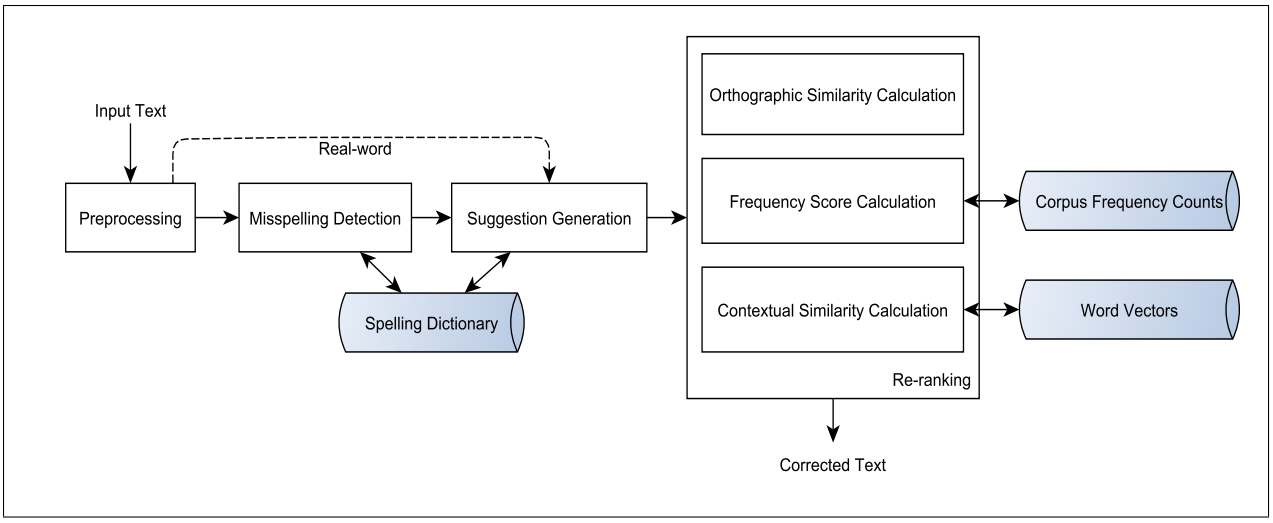

Figure 1: Spelling correction pipeline

Finally, we split tokens with leading or trailing digits, when we recognize that such a split would eliminate the spelling error (e.g., *4.5. Doctors* for *4.5.Doctors*, but not *2 nd* for *2nd*).

The second step is to detect misspellings. As most other spelling error detection systems, we use a simple dictionary lookup. Our dictionary is based on the comprehensive English dictionary that is distributed with the Jazzy spell checker[3], a open-source Java implementation based on ASpell. We expanded this dictionary with tokens extracted from terms in UMLS[23], resulting in a dictionary of 450K tokens, including inflectional variants. Needless to say, misspelling detection step skips real word errors, which are both detected and corrected in the subsequent steps.

In the third step, we generate phonetic and edit distance-based spelling suggestions. We ensure that each suggestion is a valid word with dictionary lookup. For non-word spelling errors, tokens of length two or less, and for real word errors, tokens of length three of less, are skipped, unless they can be merged with the next token to form a valid word. Phonetic suggestions are obtained using the Double Metaphone algorithm[24]. We also compute suggestions using Levenshtein distance. The maximum number of edits is taken as the minimum of the half length of the token and 5.

The next step is to rank all generated suggestions using a set of algorithms. Three of these address the orthographic similarity between the misspelling and the suggestion:

- **Token similarity**: This measure is based on the cost of converting the misspelling to the suggestion, in terms of the number of edits required. Deletion and insertion operations have a higher cost than transposition and a lower cost than substitution. For example, the token similarity scores between the misspelling *dianosed* and the suggestions *diagnosed* and *deionized* are 0.91 and 0.61, respectively. Splitting the misspelling incurs an additional penalty.

- **Phonetic similarity**: Similar to *token similarity*, it uses the phonetic representation of the misspelling and the suggestion. The phonetic similarity between *dianosed* and *diagnosed* is 1.0, while it is 0.91 between *diagnosed* and *diagnose*.

[3]http://jazzy.sourceforge.net/

- **Leading/trailing character overlap**: This measure calculates the overlap between the misspelling and the suggestion in terms of the number of matching characters at the beginning and the end. For example, the overlap score between *dianosed* and the word *diagnosed* is (3+5)/9=0.89, where the denominator corresponds to the length of the longer token.

In addition to these similarity measures, we score the suggestions by their frequency in consumer health-related corpora. To create the word frequency list, we used health-related articles in MedlinePlus Medical Encyclopedia[4], MedlinePlus Drugs[5], Genetics Home Reference[6], Genetic and Rare Diseases (GARD) frequently asked questions[7], NHLBI Health Topics[8], NINDS Disorders[9], and NIH Senior Health[10], resources targeted at the public that we use for answering health-related questions. The number of articles from each resource are given in Table 2. We ignore numbers, URLs, and email addresses. The resulting word count

Table 2: Distribution of articles used for word frequency counts and word embeddings

Resource	*Number of articles*	*Token count*
MedlinePlus Encyclopedia	4175	2,485,293
MedlinePlus Drugs	1246	1,409,389
Genetics Home Reference	1221	1,019,153
Genetic and Rare Diseases	1467	47,081
NHLBI Health Topics	140	484,449
NINDS Disorders	277	106,305
NIH Senior Health	64	219,693
TOTAL	8590	5,771,363

list consists of more than 50K words. The corpus frequency score of a suggestion is calculated as

$$freq\_score(s) = ln(C_s/N)/ln(C_{max}(w)/N)$$

where C_s is the frequency of the suggestion, $C_{max}(w)$ the maximum frequency count and N the number of tokens in the corpus, the score essentially corresponding to normalized unigram probability. Among the suggestions for *dianosed*, the score for *diagnose* is highest with 0.58; *deionized*, on the other hand, has a score of 0.

Finally, we calculate a contextual similarity score using word embeddings, taking context around the token into account. For this purpose, we used the same corpora that we used to calculate word frequency counts. We used the *word2vec* toolkit[11] with the CBOW (continuous bag-of-words) model and hierarchical softmax options with window size of 5 to generate word vectors of 200 dimensions. CBOW, unlike the traditional bag-of-words model, uses the continuous distributed representation of the context. Hierarchical softmax is a computationally efficient way to estimate the overall probability distribution. To calculate the contextual

[4]http://www.nlm.nih.gov/medlineplus/encyclopedia.html
[5]http://www.nlm.nih.gov/medlineplus/druginformation.html
[6]http://ghr.nlm.nih.gov/
[7]http://rarediseases.info.nih.gov/gard
[8]http://www.nhlbi.nih.gov/health/health-topics
[9]http://www.ninds.nih.gov/disorders/disorder_index.htm
[10]http://nihseniorhealth.gov/
[11]https://code.google.com/p/word2vec/

similarity score for the suggestion, we compute the *context vector* by averaging the vectors associated with the words in a predefined window around the token under consideration and then compute the cosine similarity between the context vector and the suggestion vector. If there is no word vector corresponding to the suggestion or if the resulting cosine similarity is less than 0, we take the contextual similarity as 0. As context, we use two tokens before and after the token in consideration. If a token in the context does not have a corresponding vector, we move to the next token in the same direction. The contextual similarity score computed in this way is 0.69 for *diagnosed* for the context given in Example 1 and 0.51 for *diagnose*.

Scores calculated by these methods are then combined using a linear-weighted ensemble, similar to Flor and Futagi[12], taking the suggestion with the highest score as the correct spelling. Using the training set, we empirically determined the best weights to be 0.6 for orthographic similarity, 0.15 for contextual similarity, and 0.25 for frequency score.

For real word spelling errors, the methodology is essentially the same. We mentioned above that we skip valid tokens with three or less characters, unless they can be merged with the next token to form another valid token. In addition, for a valid token to be considered a real word error, we stipulate that: a) the difference between the contextual similarity of the suggestion and that of the original token be greater than a threshold value (taken as 0.2 in our experiments), b) the Levenshtein distance between the suggestion and the original token be equal to or less than 2, and c) the suggestion not be an inflectional variant of the original token and vice versa.

3.3 Evaluation

We used 372 questions for training and the remaining 100 for testing. As an informed baseline, we used the ESpell algorithm[7], designed for biomedical queries and used in the PubMed search engine. For non-word error correction, we used the Jazzy spell checker to filter out correctly spelled words for both ESpell and our ensemble method. For real-word correction, no such filtering was performed. We assessed the effect of preprocessing and each of the similarity measures on the system performance. We used precision, recall, and F_1 score as the evaluation metrics.

4 Results and Discussion

The evaluation results obtained with the system are given in Table 3. Considering the non-word spelling errors only, the baseline, ESpell with non-word tokens as input, yields a F_1 score of 0.29, whereas the best weighted combination of ranking methods yields a score of 0.61. The preprocessing module is able to correct a significant number of spelling errors, providing a solid foundation which the ranking methods improve on. Orthographic similarity seems to contribute most to the prediction of the correct spellings, providing a 12% increase in F_1 score alone. Ignoring orthographic similarity and using only contextual similarity or corpus frequency diminishes the results. On the other hand, each of these methods complement orthographic similarity and together yield better precision and recall, pushing the overall F_1 score to 0.61. When the system attempts to correct real word errors as well, the baseline, in this case ESpell without any filtering step, yields a F_1 score of 0.25. This score is improved by more than two-fold to 0.58, when we use the setup that yields best results for non-word spelling errors with additional constraints. When we take into account only the spelling errors that were deemed to be important for focus or frame extraction, we obtain a F_1 score of 0.76, in comparison to the baseline score of 0.57.

The corrections made with preprocessing may seem unimportant; however, we believe these corrections

Table 3: Evaluation results

Method	*Precision*	*Recall*	F_1
Non-word only			
ESpell (with filtering)	0.53	0.20	0.29
Preprocessing only	0.94	0.33	0.49
W/ Orthographic similarity	0.57	0.52	0.55
W/ Corpus frequency	0.45	0.41	0.43
W/ Context similarity	0.42	0.38	0.40
ALL	0.64	0.58	0.61
Real-word included			
ESpell	0.23	0.26	0.25
ALL	0.57	0.59	0.58
Important for focus/frame only			
ESpell	0.58	0.56	0.57
ALL	0.83	0.70	0.76

will contribute to better tokenization and parsing, ultimately leading to better performance in information extraction. The results confirm that orthographic similarity is the best predictor of correct spellings. In general, edit distance of 2 or 3 is considered sufficient for successful spelling correction, based on the observation of Damerau[10] that 80% of spelling errors are caused by a single edit. Our dataset provides evidence that spelling correction in consumer health questions may require considering suggestions with a higher edit distance (for example, in Example 1, *seretona* has an edit distance of 3 from its correct spelling, *serotonin*). This may be partly due to the fact that NLM receives questions from non-native speakers and partly due to the complexity of spelling of long medical terms.

We find that corpus frequency and contextual similarity contribute to spelling correction, though not as significantly as orthographic similarity. This may be partly due to the coverage of the corpora we considered. Their total size (approximately 5.8M tokens) is small, relative to the much larger corpora often employed in spelling correction. We experimented with larger corpora, such as one based on Wikipedia articles; however, we found that they did not make a significant difference, possibly because they are better written than health questions and are not health information-specific. To calculate contextual similarity, we use word embeddings, which, to our knowledge, is the first application of this technique to spelling correction. While the performance improvement due to word embeddings was relatively small, we believe it is a promising avenue to explore further. In its application to spelling correction task, one open question is how to best calculate contextual similarity using word vectors. We experimented with several measures and found that cosine similarity between the spelling suggestion and the average context vector yielded the best result; however, other measures (e.g., the maximum cosine similarity between the suggestion and a vector for any of the words in context) or a combination thereof could prove more beneficial.

The system, not unexpectedly, has more difficulty with real word spelling errors than with non-word errors. Our pipeline for real word errors differs little from that for non-word errors; it only incorporates several additional constraints. The results indicate that a more nuanced approach, probably with a different weighting scheme, may be necessary. It is worth noting that most systems focusing solely on real word spelling errors build on specific confusion sets[15,18] or datasets to which real word errors are artificially introduced[13,16],

and assume that the context around the error contains no spelling errors. We did not make such assumptions for the real word errors in our somewhat noisy dataset, and thus a fair comparison may be difficult.

Using a comprehensive, well-curated dictionary is critical for identifying misspellings. To have a better coverage of the biomedical domain, we extended an existing general English dictionary automatically with term tokens from UMLS, resulting in a comprehensive, albeit somewhat noisy, dictionary. Some errors were due to the noise in the dictionary. For example, the token *Peyronies* was simply skipped, since the word existed in the dictionary, although its canonical form, and the correct spelling in our dataset, is *Peyronie's*.

Misspelling of adjacent words is a well-recognized problem in spelling correction and we observed errors due to this problem, as well. In Example 1, we are able to correct *on set* to *onset* based on contextual similarity when the following token is corrected from *deminita* to *dementia*; otherwise, we get a recall error.

5 Conclusion

We presented a spelling correction system for consumer health questions that takes into account orthographic features as well as corpus frequency information and contextual similarity. We did not attempt to use any semantic information (e.g., UMLS) beyond that obtained in an unsupervised manner from corpora. Our results show that all the components contribute to non-word spelling error correction, while real word errors remain challenging. Since the system is designed as part of a larger information extraction system, it is encouraging that it is able to recognize and correct the majority of spelling errors most relevant to information extraction.

The future work involves using more suitable corpora as the basis of frequency counts and word embeddings. We believe that a corpus that specifically focuses on text from online health forums would be more appropriate, since the questions we receive are probably closest to such text. We are currently incorporating such a corpus into our system. We are also planning to explore ways to better rank the similarity scores, probably in a machine learning framework. Another possible direction is to explore joint spelling correction, since misspelling of adjacent words is common in consumer health questions.

Acknowledgments

This work was supported by the intramural research program at the U.S. National Library of Medicine, National Institutes of Health.

References

[1] Zhang Y. Contextualizing consumer health information searching: an analysis of questions in a social Q&A community. In: Proceedings of the 1st ACM International Health Informatics Symposium; 2010. p. 210–219.

[2] Roberts K, Kilicoglu H, Fiszman M, Demner-Fushman D. Decomposing Consumer Health Questions. In: Proceedings of the 2014 Workshop on Biomedical Natural Language Processing; 2014. p. 54–62.

[3] Roberts K, Kilicoglu H, Fiszman M, Demner-Fushman D. Automatically Classifying Question Types for Consumer Health Questions. In: AMIA Annual Symposium Proceedings; 2014. p. 1018–1027.

[4] Kilicoglu H, Fiszman M, Demner-Fushman D. Interpreting Consumer Health Questions: The Role of Anaphora and Ellipsis. In: Proceedings of the 2013 Workshop on Biomedical Natural Language Processing; 2013. p. 54–62.

[5] Klein D, Manning CD. Accurate unlexicalized parsing. In: Proceedings of the 41th Meeting of the

Association for Computational Linguistics; 2003. p. 423–430.

[6] Mikolov T, Chen K, Corrado G, Dean J. Efficient Estimation of Word Representations in Vector Space. CoRR. 2013;abs/1301.3781. Available from: http://dblp.uni-trier.de/db/journals/corr/corr1301.html#abs-1301-3781.

[7] Wilbur WJ, Kim W, Xie N. Spelling correction in the PubMed search engine. Information Retrieval Boston. 2006;9(5):543–564.

[8] Kukich K. Techniques for automatically correcting words in texts. ACM Computing Surveys 24. 1992;p. 377–439.

[9] Mitton R. Ordering the suggestions of a spellchecker without using context. Natural Language Engineering. 2009;15(2):173–192.

[10] Damerau FJ. A Technique for Computer Detection and Correction of Spelling Errors. Commun ACM. 1964;7(3):171–176.

[11] Levenshtein V. Binary Codes Capable of Correcting Deletions, Insertions and Reversals. Soviet Physics Doklady. 1966;10:707.

[12] Flor M, Futagi Y. On using context for automatic correction of non-word misspellings in student essays. In: Proceedings of the Seventh Workshop on Building Educational Applications Using NLP; 2012. p. 105–115.

[13] Hirst G, Budanitsky A. Correcting real-word spelling errors by restoring lexical cohesion. Natural Language Engineering. 2005;11(1):87–111.

[14] Fellbaum C. WordNet: an electronic lexical database. Cambridge, MA: MIT Press; 1998.

[15] Golding AR, Roth D. A Winnow-Based Approach to Context-Sensitive Spelling Correction. Machine Learning. 1999;34(1-3):107–130.

[16] Islam A, Inkpen D. Real-Word Spelling Correction using Google Web 1T 3-grams. In: Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing; 2009. p. 1241–1249.

[17] Whitelaw C, Hutchinson B, Chung GY, Ellis G. Using the Web for Language Independent Spellchecking and Autocorrection. In: Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 2 - Volume 2. EMNLP '09; 2009. p. 890–899.

[18] Xu W, Tetreault JR, Chodorow M, Grishman R, Zhao L. Exploiting Syntactic and Distributional Information for Spelling Correction with Web-Scale N-gram Models. In: EMNLP; 2011. p. 1291–1300.

[19] Crowell J, Zeng Q, Ngo LH, Lacroix EM. A Frequency-based Technique to Improve the Spelling Suggestion Rank in Medical Queries. JAMIA. 2004;11(3):179–185.

[20] Ruch P, Baud R, Geissbühler A. Using lexical disambiguation and named-entity recognition to improve spelling correction in the electronic patient record. Artificial intelligence in medicine. 2003;29(1-2):169–184.

[21] Patrick J, Sabbagh M, Jain S, Zheng H. Spelling correction in Clinical Notes with Emphasis on First Suggestion Accuracy. In: Proceedings of 2nd Workshop on Building and Evaluating Resources for Biomedical Text Mining (BioTxtM2010). Malta; 2010. p. 1–8.

[22] Bengio Y, Ducharme R, Vincent P, Jauvin C. A Neural Probabilistic Language Model. Journal of Machine Learning Research. 2003;3:1137–1155.

[23] Lindberg DAB, Humphreys BL, McCray AT. The Unified Medical Language System. Methods of Information in Medicine. 1993;32:281–291.

[24] Philips L. The Double Metaphone Search Algorithm. C/C++ Users Journal. 2000;18(6):38–43.