ELSEVIER Artificial Intelligence in Medicine 29 (2003) 169–184

Artificial Intelligence in Medicine

www.elsevier.com/locate/artmed

Using lexical disambiguation and named-entity recognition to improve spelling correction in the electronic patient record

Patrick Ruch[a,b,*], Robert Baud[a], Antoine Geissbühler[a]

[a]Medical Informatics Division, University Hospital of Geneva, 1205 Geneva, Switzerland
[b]Theoretical Computer Science Laboratory, Swiss Federal Institute of Technology, 1012 Lausanne, Switzerland

Received 7 January 2002; received in revised form 11 April 2003; accepted 12 April 2003

Abstract

In this article, we show how a set of natural language processing (NLP) tools can be combined to improve the processing of clinical records. The study concentrates on improving spelling correction, which is of major importance for quality control in the electronic patient record (EPR). As first task, we report on the design of an improved interactive tool for correcting spelling errors. Unlike traditional systems, the linguistic context (both semantic and syntactic) is used to improve the correction strategy. The system is organized along three modules. Module 1 is based on a classical spelling checker, it means that it is context-independent and simply measures a string-edit-distance between a misspelled word and a list of well-formed words. Module 2 attempts to rank more relevantly the set of candidates provided by the first module using morpho-syntactic disambiguation tools. Module 3 processes words with the same part-of-speech (POS) and apply word-sense (WS) disambiguation in order to rerank the set of candidates. As second task, we show how this improved interactive spell checker can be cast as a fully automatic system by adjunction of another NLP module: a named-entity (NE) extractor, i.e. a tool able to identify words as such patient and physician names. This module is used to avoid replacement of named-entities when the system is not used in an interactive mode. Results confirm that using the linguistic context can improve interactive spelling correction, and justify the use of named-entity recognizer to conduct fully automatic spelling correction. It is concluded that NLP is mature enough to help information processing in EPR.

Keywords: Electronic patient record; Named-entity recognition; Spelling correction; Natural language processing; Artificial intelligence

[*] Corresponding author. Tel.: +41-22-372-61-64; fax: +41-22-372-80-60.
E-mail address: patrick.ruch@dim.hcuge.ch (P. Ruch).

0933-3657/$ – see front matter © 2003 Elsevier Science B.V. All rights reserved.
doi:10.1016/S0933-3657(03)00052-6

1. Introduction

Spelling correction in clinical texts constitutes a critical issue, because misspellings will likely cause dramatic side effects (see for example [17,18]). These studies, mostly conducted on medical orders, conclude that automated measures of similarities between medication names can form the basis of highly accurate, sensitive, and specific tests of the potential for errors with look-alike and sound-alike medication names. We assume that the importance of reducing misspellings can be more or less applied to all the content of patient files, so that the quality of documents dealing with other aspects of clinical narratives (anamnesis, findings, diagnosis, laboratory and test results ...) should be carefully controlled as well.

Some recent studies [40] confirm that misspellings are rather frequent in medical web queries, whereas in a comparative corpus linguistic study [34], we reported that rates of misspellings in patient records—up to 10% in follow-up notes—are significantly higher than in other corpora, such as journal articles. Indeed journal samples are to be read by a large public, and so are more carefully written than documents in the patient file, which are supposed to remain in the hospital institution.[1] At this time we concluded that using journal abstracts, as for example, MedLine corpora, to assess the impact of natural language processing (NLP) tools could result in the construction of systems poorly suited for clinical narratives. In parallel, it has been reported that document corruptions (due to misspellings or introduced by optical character recognition) can largely affect performances of text processing tools, such as information retrieval engines [15,32],[2] therefore misspellings are a priority for those concerned with text engineering in clinical applications.

Let us observe that due to the poor performance of existing tools, spelling correction is usually performed in interaction with the user, who is asked to validate the proposition of the corrector. In these tools, an important source of erroneous propositions comes from the presence of named-entities (NE) such as person names, which are often classified as misspelled words because they are not listed in the terminological resources of the spelling tool. In order to provide a non-interactive spelling corrector it is necessary to first identify such words. For a general introduction on named-entities extraction, we shall refer to the message understanding conferences (MUC).[3] A practical application of such tools is reported in [38], where the authors present a NE "scrubber", i.e. a system able to remove confidential items from clinical documents to solve confidentiality issues.

The remainder of this article is organized as follows. Section 2 gives an overview of past and current researches in spelling correction and related fields. In Section 3, we describe the development of our tools: first, we present the improved interactive corrector; and second, we show how this tool can be adapted to conduct non-interactive spelling

[1] This is not true for reports such as discharge summaries.

[2] Information retrieval engines are for example used for case-based reasoning [29].

[3] See the NE task definition in the MUC/TREC framework: http://www.cs.nyu.edu/cs/faculty/grishman/Netask20.book_1.html. An excellent overview on Information Retrieval and Extraction in biomedical corpora can be found in [24].

correction tasks via the addition of a named-entity recognizer. In Section 4, we provide an evaluation of each system. A conclusion follows in Section 5.

2. Spelling correction overview

Spelling correction problems can be separated in two categories. The first category addresses the problem of correcting spelling that result in valid, though unintended words (as for example,[4] in *a peace of cake*, where *piece* is misspelled) and also the problem of correcting particular word usage errors (such as *among* and *between*). The second category is concerned only with errors that result in non-existent words, i.e. words that cannot be found in a lexicon. While the first problem is sometimes referred to as *context sensitive spelling correction*, with numerous recent studies (see for example [10,11,20]), the second, referred implicitly *as context-free spelling correction*, is perceived as a problem where progress cannot be made [27],[5] however, some works show the importance of the context for improving accuracy of the second category too, as in [4,21].

At this level, we suggest a new terminology for qualifying each category: we will call *word correction* the first category, and *character correction* the second, leaving the *context sensitivity/insensitivity* question open for both correction types. Although the linguistic approach we adopt in this study could be applied to both types of spelling errors, we concentrate on errors resulting in non-existent words, which are far more frequent than errors resulting in valid words.

2.1. Lexical coverage and open class words

As we are interested in correcting misspellings that result in non-existent words, the first part of the work consists in collecting a large set of well-spelled tokens. This set will be used to decide whether a given word is well spelled or not, and will constitute the main lexical resource of our tool. Leaving aside the collecting process, the second immediate problem concerns the comprehensiveness. Indeed, there exist some particular classes of words that cannot be listed in any finite lexicon: as for example proper names, names of location (towns, villages, rivers . . .). These words are referred to as named-entities and constitute a major problem when building a non-interactive spell checker. Thus, when trying to spell check a discharge summary with a standard interactive system, it appears that the vast majority of words, which are signaled as misspelled, are in fact named-entities. The reason why NE recognition (NER) is not performed on top of the spelling correction is because such entities do not disturb the human-assisted correction. But, if these entities represent a trivial noise for a human user, they definitely constitute a major challenge for a system supposed to perform the correction task in a batch mode.

[4] The experiments were conducted on French corpora, however when possible, examples are provided in English for sake of clarity.

[5] However, the problem is still very crucial for agglutinative languages [26], where the vocabulary can be hardly listed in a comprehensive manner. Although some authors [1] underlined the high compositionally of the medical language even for morphologically poor languages such as French and English, the French medical language will be considered fully listed in a 200,000 entry list of words.

2.2. Part-of-speech disambiguation

While recent experiments on word correction application use some linguistic modules (mainly part-of-speech (POS)[6] disambiguation tools as in [11]) for handling the context, we observe that most character correction tools—even when they use the context—do not use comparable approaches, and instead rely on simple word language models [4,21]. The first specificity of our system consists in applying morpho-syntactic disambiguation to the character correction problem.

Working on a word correction problem, Golding and Shabes [11] introduce a method using POS trigrams[7] to encode the context. Although this method greatly reduces the number of parameters compared to methods based on word trigrams,[8] it empirically appeared to discriminate poorly when words in the confusion set have the same POS. In this last case, the method is coupled with a more traditional word model. Like them, we start filtering with a POS tagger, but then, we also explore the use of a word-sense (WS)[9] tagger for discriminating among candidates, which have the same POS. Here is located the second specificity of the approach.

2.3. Syntactic correction

Syntactic correction is a different but related task. Syntactic correction addresses (a) word order/presence, and (b) agreement problems:

a. He wants play tennis *to*./He wants to play tennis.
b. They picks a piece of cake./They pick a piece of cake.

Of course, character correction and word correction are necessary to provide a correct syntactic correction, therefore in such systems, the usual processing is more complex: first, a string-edit module solves the character errors; and second a syntactic module looks for syntactic errors [7]. Here lies the third specificity of our approach as we apply morpho-syntactic constraints at the character correction level.

2.4. String-to-string edit-distance

Another promising way of research attempts to improve the string-edit-distance module (as explored in [4,6,30]; see [14] for recent survey). However, this promising research path

[6] In English, a part-of-speech is a syntactic—also called morpho-syntactic—category, as for example: noun, verb, adjective, determiner, and adverb.

[7] *N* grams are an ordered set of *N* consecutive items.

[8] POS *n* grams represent the morpho-syntactic level, word *n* grams represent the token level, and word-sense *n* grams the semantic level, thus the phrase *he diagnoses* can be represented by three different models: *he diagnoses* (word level), *prop v[03]* (morpho-syntactic level), and *pers diap* (semantic level). The meaning of prop, v[03], pets, and diap is, respectively, personal pronoun, verb third person, *human being* (UMLS TO16), and *diagnostic procedure* (UMLS T060). POS tags attempt to follow the MULTEXT morpho-syntactic description (http://www.lpl.univ-aix.fr/projects/multext/).

[9] The semantic classes are based on the UMLS Semantic Network, with classes such as body part, diseases, and temporal concept.

requires large amounts of training data (as for example [30], who worked with a three million word corpus for speech recognition) that are often absent apart from some very particular sets of domains and languages, and anyway absent for the French medical sublanguage. Therefore in our experiments, we use a rather basic edit-distance known as the Levenshtein–Damerau distance (see Section 3.2).

3. Method: Balancing Act

The improved interactive spelling checker relies on three modules, which are applied in a serialized manner:[10]

- the first module is based on a context-independent string-to-string edit-distance (StrEditDist) calculus;
- the second module, based on the morpho-syntactic context, attempts to rerank the data set provided by the first module;
- the third module processes words with the same POS by applying contextual WS disambiguation.

The non-interactive tool uses a NE recognizer on top of these modules to preprocess NE. In our system, the morpho-syntactic and semantic filtering can be seen has a winner-takes-all process, where only the most reliable part-of-speech is given more weight, similarly to what occurs in a decision-list system [31]. The taggers combine handcrafted regular rules, and Hidden Markov Models (HMM; for processing the remaining ambiguities) to select the most reliable part-of-speech or word-sense candidates.

Unlike standard Bayesian approaches, however, such an approach does not combine the log-likelihood of each classifier, but bases its classification solely on the most reliable piece of evidence identified in the target context. Perhaps surprisingly, this approach provides the same or even slightly better precision than approaches based on combining log-likelihoods [39], however, its major advantage is maybe to gather multiple heterogeneous classifiers, operating on non-independent source of evidence, in a unified and traceable framework. Such an architecture seems particularly well adapted in the context of developing a spelling checker tailored for medical texts, using both symbolic constraints and data-driven source of evidence, together with facing sparse data issues.

Indeed, we use a hybrid part-of-speech tagger: first, it applies a set of regular rules and then, a Hidden Markov Model to solve the remaining ambiguities. Some detailed evaluations [35,37] showed that the system has a disambiguation rate above 96% for medical corpora. As for the word-sense disambiguation task, it behaves along the same lines as the POS tagger, with similar performance, and has also been recently described together with its 40 UMLS-based semantic types [3,36].

[10] *Balancing Act* refers to hybrids methods in natural language processing, which attempt to balance expert linguistic knowledge and computational learning methods, as suggested in [16].

Table 1
Example of records in the evaluation database

ID	Left context 2	Left context 1	Misspelled word	Right context 1	Right context 2	Well-formed word
77	Evidence	un	Oedéme	sous	cutané	oedéme
78	Une	paroi	Souileé	avec	présence	souillée
79	Par	le	Sphinter	on	introduit	sphincter

3.1. Evaluation database

For evaluation purposes, we collected a set of misspelled words together with the left and right adjacent context (two words before and two words after the misspellings), and we got a total of 424 records. This initial set is split in two equal subsets (212 records), set A is used for tuning the system, while set B is kept as final test set. This collection step was done manually.[11] Table 1 gives some examples of the misspelled words (in French).

The main lexical resource is a 200,000 item list of well-written tokens and it is used by the string-to-string edit-distance module. As additional resource, a 30,000 lexemes lexicon is necessary for POS filtering [1]. In these lexemes about 20% are provided with a semantic type (WS tag) to be used for WS filtering purposes.

3.2. String-to-string edit-distance calculus

Modem spelling checkers[12] are usually based on a variant of the Levenshtein–Damerau distance, which computes—via dynamic programming—the number of operations for transforming a string into another one. Thus, in Fig. 1, the distance between *liver* and *livver* is 1, because a unique insertion is sufficient to transform the string *liver* into the string *livver*, and the cost associated with such an operation is 1 (as defined in Fig. 1). Damerau [8] indicates that 80% of all spelling errors are the result of:

a. transposition of two adjacent letters: he*ap*titis (err1);
b. insertion of one letter: hep*p*atitis (err2);
c. deletion of one letter: hepattis (err3);
d. replacement (deletion + insertion) of one letter by another one: hepatotis (err4).

In the standard model, which is used in the example given in Fig. 1, each of these operations cost 1, i.e. the distance between err1, err2, err3, err4 and the word *hepatitis* is 1, while the distance between *hepatitis* and *heppatotis* is 2 (one replacement + one insertion). But more accurate models have been developed, where each operation might have an associated cost, depending on the left adjacent character [6].

[11] Corpora for spelling correction are rare in French. The misspelled word database we collected will become available as additional package of the FRIDA resources: http://www.fltr.ucl.ac.be/fltr/germ/etan/cecl/Cecl-projects/Frida/frida.htm.

[12] Alternative approaches include *n* gram distances and similarity keys, cf. [28].

		-1	0	1	2	3	4	5
			l	**i**	**v**	**v**	**e**	**r**
-1		0	1	2	3	4	5	6
0	**l**	1	0	1	2	3	4	5
1	**i**	2	1	0	1	2	3	4
2	**v**	3	2	1	0	1	2	3
3	**e**	4	3	2	1	1	1	2
4	**r**	5	4	3	2	2	2	**1**

Given an alphabet Σ, three different string operators are defined: substitute, delete and insert. In the above diagram, values associated with each of these operations are the following:

Substitute (*a*,*a*) = 0, for a ∈ Σ;
Substitute (*a*,*b*) = 1, for a,b ∈ Σ and a ≠ b;
Delete (*a*) = *Insert* (*a*) = 1, for a ∈ Σ;

Fig. 1. Computation of the Levenshtein distance between *liver* and *livver*.

The error model we developed includes some minor refinements as compared to the model given in Fig. 1. Thus, if the default replacement operation has a one unit cost, some more probable replacements (in French, a frequent confusion is for example the letter set {ï, i}) will be less expensive. The cost matrix was tuned manually by applying regression tests on the set A. Indeed, considering the size of the set A, maximization expectation methods (as in [30]) were hardly applicable here.

3.3. Contextual filtering

After processing by the edit-distance module, each candidate word comes out with a score. This score expresses the distance between the candidate and the misspelled word. The two following modules are applied sequentially to get a more optimal ranking of the candidates. It is important to notice that if one word within the candidate set is not provided with a POS tag, then following filters (POS and WS) are not applied. Similarly, the WS filter is not applied if one of the candidates is provided without WS tag. This caution is important in order not to favor words listed in our word-sense lexicon versus words appearing in the 200,000 items list.

3.3.1. Part-of-speech filtering

The goal of this module is to modify the edit-distance scoring by considering the morpho-syntactic information. In Fig. 2, let us consider a misspelled word in context, together with a short list of likely candidates. List 1 provides the list as returned by the MS-Word 2000 English spell checker, while list 2 shows what our system returns by taking advantage of the left context.

In this example, it is observed that list 2 provides a more accurate ranking than list 1 for the misspelled string *uncer*. Indeed, if we consider the adjacent left context: a determiner such as *an* cannot be followed by a preposition such as under.

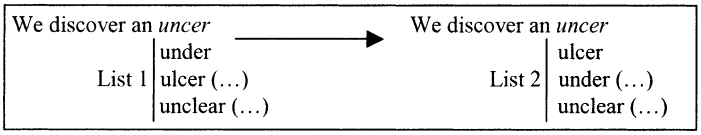


Fig. 2. Example of part-of-speech filtering.

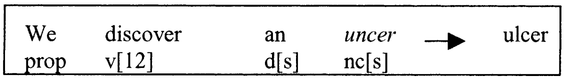


Fig. 3. POS disambiguation (after POS tagging).

The POS tagger attempts to attribute one part-of-speech to every token. Thus, for the above example, and after a lexical access (Fig. 4) the tool provides the choice given in Fig. 3. The meaning of the POS tags follows:

- prop: personal pronoun;
- v[12]: verb first or second person;
- d[s]: determiner singular;
- nc[s]: common noun singular;
- sp: preposition;
- a: adjective.

In the above figures, the candidate (there could be more than one) with the tag *nc[s]*, which is the most likely POS tag in this context, is favored.

3.3.2. *Word-sense filtering*

There are traditionally two ways to process the semantic filtering: implicitly, for example, by working with word language models, or explicitly by using syntactic and semantic representation levels. The semantic representation used in this module capitalizes on the 134 semantic types and 54 relationships of the UMLS Semantic Network [25]. But like [22], we define a computationally more manageable subset of these semantic classes (see [36], for a detailed report).

In Fig. 5, the part-of-speech does not provides any discrimination rule between the candidates, as both have the same part-of-speech (*nc[s]*). However, the semantic left adjacent context can operate as a discriminator, indeed *incise* (tagged *thers*; see below for the meaning of WS tags) is to be followed by some body part (*loc*), like for example *liver*,

We	discover	an	*uncer*
prop	*v[12]*	*d[s]*	*sp* *nc[s]* *a*

Fig. 4. POS lexical ambiguity (before POS tagging).

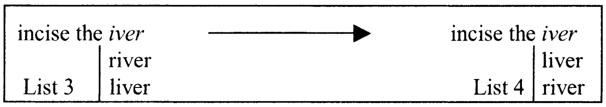


Fig. 5. Example of word-sense filtering (when applying WS tagging, stop words (determiner, auxiliary . . .) are removed).

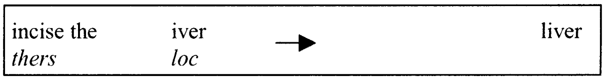


Fig. 6. WS disambiguation (after WS disambiguation).

rather than by some human general object (*obj*), which is the semantic class of river (Fig. 5). In Figs. 4–7, the meaning of WS tags is the following:

- thers: therapeutic procedure (UMLS T061);
- obj: object (UMLS T073);
- loc: organ and body location (UMLS T023/TO29).

When processing the above sentence, and after lexical access (Fig. 7), the word-sense tagger selects the output in Fig. 6.

3.3.3. Interactive spelling correction: forward and backward

In a spelling correction task, the interaction between the system and the user is possible along two different modes. First, the system can signal each misspelled word while the user is still typing; second, the user runs the system when typing is completed, and the system prompts for some suggested changes. In the first case, the system can rely on the left context, while in the second case both right and left side of every word is available. Depending on the selected mode, it is possible to design a system that is capable to take advantage of the right context as well the left one. Therefore, we also evaluate the improvement brought by a system using both contexts as compared to a half blind system.

The improvement can be illustrated by the example in Fig. 8, where the best candidate cannot be decided until the left side of the sentence has been completed (using the subject–verb agreement constraint).

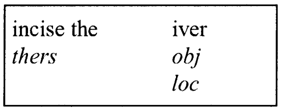


Fig. 7. WS disambiguation (before WS disambiguation).

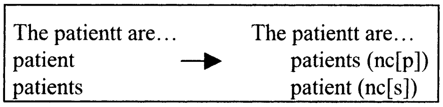


Fig. 8. Bi-directional disambiguation.

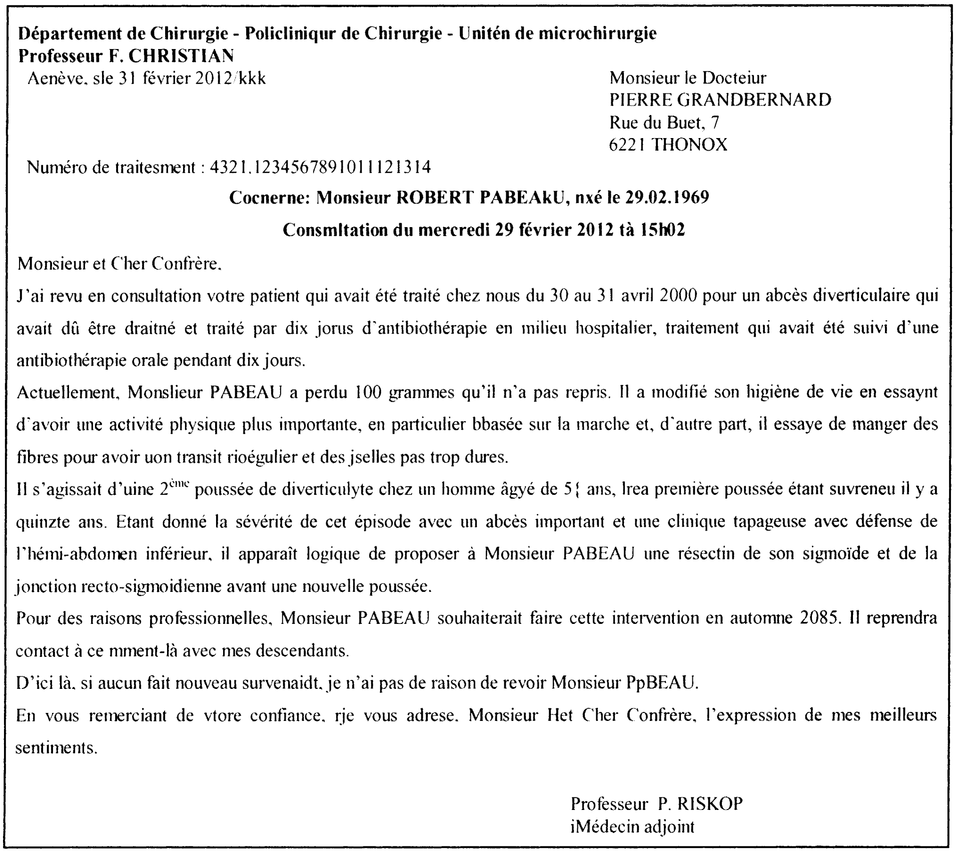
Finally, we must acknowledge that left and right contexts are necessary in some cases, but might still not provide enough context in several cases, as for example in:

*The patient*t *showed that the ...*

3.4. Toward a non-interactive spelling correction based on NE recognition

In the non-interactive mode, we use the contextual spelling corrector, with left and right context. The tool is applied on a corpus of medical documents, which has been preprocessed by the NE recognizer. As they are made explicit, these NE will be ignored by the spelling checker.

Various methods have been successfully applied to NE extraction tasks: statistical [2], rule-based [13] and hybrids [23]. We rely on a rule-based system, in the spirit of [13]. See [33] for a detailed presentation, where the recognition system is tailored as a scrubbing



Département de Chirurgie - Policliniqur de Chirurgie - Unitén de microchirurgie
Professeur F. CHRISTIAN
Aenève. sle 31 février 2012/kkk

Monsieur le Docteiur
PIERRE GRANDBERNARD
Rue du Buet, 7
6221 THONOX

Numéro de traitesment : 4321.1234567891011121314

Cocnerne: Monsieur ROBERT PABEAkU, nxé le 29.02.1969

Consmltation du mercredi 29 février 2012 tà 15h02

Monsieur et Cher Confrère.

J'ai revu en consultation votre patient qui avait été traité chez nous du 30 au 31 avril 2000 pour un abcès diverticulaire qui avait dû être draitné et traité par dix jorus d'antibiothérapie en milieu hospitalier, traitement qui avait été suivi d'une antibiothérapie orale pendant dix jours.

Actuellement, Monslieur PABEAU a perdu 100 grammes qu'il n'a pas repris. Il a modifié son higiène de vie en essaynt d'avoir une activité physique plus importante, en particulier bbasée sur la marche et, d'autre part, il essaye de manger des fibres pour avoir uon transit rioégulier et des jselles pas trop dures.

Il s'agissait d'uine $2^{\text{ème}}$ poussée de diverticulyte chez un homme âgyé de 5{ ans, lrea première poussée étant suvreneu il y a quinzte ans. Etant donné la sévérité de cet épisode avec un abcès important et une clinique tapageuse avec défense de l'hémi-abdomen inférieur, il apparaît logique de proposer à Monsieur PABEAU une résectin de son sigmoïde et de la jonction recto-sigmoidienne avant une nouvelle poussée.

Pour des raisons professionnelles, Monsieur PABEAU souhaiterait faire cette intervention en automne 2085. Il reprendra contact à ce mment-là avec mes descendants.

D'ici là, si aucun fait nouveau survenaidt, je n'ai pas de raison de revoir Monsieur PpBEAU.

En vous remerciant de vtore confiance. rje vous adrese. Monsieur Het Cher Confrère, l'expression de mes meilleurs sentiments.

Professeur P. RISKOP
iMédecin adjoint

Fig. 9. Corrupted document sample (the authors have largely modified the original document, so that facts as well as confidential items are pure fiction).

Table 2
Baseline measures: minimal edit-distance and word model

	StrEditDist (%)	+Word (%)
T-1	89.4	93.0
T-2	93.7	96.5
T-3	96.9	97.9
T-4	97.4	98.5
T-5	97.9	99.0
T-7	98.3	99.0
T-20	99.0	99.0

system, i.e. a system for removing confidential items (phone, date of birth, names . . .) in patient records, with a precision ranging from 96 to 99%.

As for evaluation corpora, we randomly selected a set of 50 discharge summaries (14,932 tokens) from the digestive surgery unit, but only a sample (2647 tokens) was used in the evaluation (Section 4.2). This type of reports was selected due to its lexical richness, and because NEs are particularly frequent in such documents. Indeed administrative headers, including patient and health care professionals identities, as well as signatures and copies are excellent NEs sources.

We corrupted the corpus at a 15% rate (see Fig. 9, for a representative example): it means that we randomly introduce a spelling error every 6.66 words. To do so, we design a corruption model, consistent with the Damerau's distribution. As 80% of misspellings can be generated from a correct spelling by four simple rules (given in Section 3.2), we applied one of the four operations to 80% of the corpus. Then, we introduced via a second process another 20% of errors, which could not be produced with such a model, mainly for approximating sound-alike errors (as for example, the character *i* is more likely to be replaced by the character *y*, than by a *q*).

4. Results

In the first experiment, we assess the correction effectiveness of the tool. In the second experiment we assess the effect of adding a NE recognizer to avoid the replacement of NEs.

4.1. Interactive spelling correction

For the evaluation, only one test run was performed on set B. In Table 2, correction strategies based on a string-to-string edit-distance calculus, augmented with a trigram word language model (Word[13]), is taken as a baseline to evaluate the improvement brought by linguistically motivated approaches.

In these experiments (Tables 2–4), we do not set any confidence threshold (i.e. the maximal number of operations for transforming a string into another one) for the string distance module; therefore the list of candidates returned by the system is never empty.

[13] In Tables 3 and 4 the word language model uses only the left context.

Table 3
Contextual spelling correction results with the left context

	POS (%)	POSWord (%)	WS
T-1	95.7	96.7	96.4
T-2	97.4	98.1	97.6
T-3	98.3	98.6	98.3
T-4	98.8	99.0	98.8
T-5	99.0	99.0	99.0

Table 3 provides results when using the left context, so only fields "Left context 1" and "Left context 2" from the evaluation DB are available. T-1 gives the precision of the system for the top candidate (recall = 1). T-2 gives the precision of the system for recall = 2 candidates, etc. Table 4 provides results similar to Table 3, but using both left and right contexts, it means that fields "Right context 1" and "Right context 2" are also used. In Tables 3 and 4, the combination of POS and Word filtering is done sequentially: the list of candidates is first ranked using the word language model, before being ranked using part-of-speech evidences. The meaning of the abbreviations in Tables 2–4 follows:

- Word: StrEditDist + word language filtering;
- POS: StrEditDist + part-of-speech filtering;
- WS: StrEditDist + part-of-speech filtering + word-sense filtering;
- POSWord: POS + word language.

The first observation concerns the central role of the contextual filter: in comparison to the basic string-edit output (StrEditDist), results are always improved by the addition of a contextual classifier (Word, POS, WS or POSWord).

The combination part-of-speech + word language (POSWord in Tables 3 and 4) generally outperforms the other combinations, and a context based both on the left and the right always improves significantly the correction rate as compared to a context based exclusively on the left context. Thus, score of POSWord with left and right contexts (Table 4) improves POSWord with left context (Table 3) with 97.2 versus 96.7, respectively, for T-1.

It is observed that word-sense filtering brings little improvement as compared to using a purely statistical word language model; the lexical sparseness of the semantic annotation is probably the main explanation for such disappointing results. More generally the scalability of the semantic module is questionable. While broad-coverage syntactic lexicons are now available for most European languages, the availability of semantic lexical resources is mostly limited to the English language [25].

Table 4
Contextual spelling correction results with left and right contexts

	POS (%)	POSWord (%)	WS
T-1	96.4	97.2	98.4
T-2	98.5	98.8	98.5
T-3	98.6	98.8	98.6
T-4	98.8	99.0	98.8
T-5	99.0	99.0	99.0

The contrast between the context-independent string-to-string distance strategy (StrEditDist in Table 2) and the approach relying on the largest linguistic context (POSWord in Table 4) is maximal for T-1: 7.8%, but the improvement is also observed for other recall values, up to T-20 (in Table 2). It is also interesting to explore the origin of the errors in Table 4. The remaining error rate for T-1 using the POSWord context is largely caused by short misspellings: they contribute for 83% (10 misspellings) of the error rate (2.8%, which represents 12 misspellings). Thus, the misspelled word *raee* generates 16 candidates: *rasée*, *ratée*, *rayée*, *race*, *rade*, *rage*, *raie*, *rame*, *rare*, *rase*, *rate*, *ré*, *râle*, *rasé rayé*, *ruée*. Hopefully these short words account for a manageable fraction of the lexicon, therefore a possible improvement would be to develop special strategies for correcting short words: we could for example calculate a more accurate language model that would include a set of examples tailored for these words. Finally, results of the combination POS + word language must be emphasized if we consider that two misspelled words (i.e. about 1% of set B) were absent from the terminological resources, so that the maximum theoretical score is 99% for this experiment.

4.2. Fully automatic spelling correction

In this evaluation (Table 5), we simulate an automatic spelling correction task. First, we evaluate the spelling checker without preprocessing named-entities (used as baseline), and second with preprocessing named-entities. Unlike in previous experiments, which were conducted without any threshold value, the string-edit-distance is now computed with two different thresholds. This threshold is important to block the correction when the distance between the top returned candidate and the misspelled word is too large. Two runs are performed:

- Run 1: threshold 1.5, which corresponds roughly to one basic operation + one operation on diacritics;
- Run 2: threshold 3, which corresponds roughly to two basic operations + two operations on diacritics.

Table 5
Results of contextual spelling correction with pre-processing of NEs

	Threshold 1.5		Threshold 3	
	Baseline (%)	With NER (%)	Baseline (%)	With NER (%)
(1) Words (strings separated by blanks)	2647			
(2) Corrupted strings (%)	361 (13.6)			
(3) Nes (%)	98 (3.7)			
(4) Non-lexical strings (%; NE + misspelled word + missing word)	468 (17.7)			
(5) Successful correction (%)	334 (92.5)	334 (92.5)	343 (95.0)	343 (95.0)
(6) Correction errors (%; including correction of Nes)	102 (21.8)	12 (2.6)	110 (23.5)	17 (3.6)

Statistics in rows 1–4 are common to each system (with and without NE recognition). In row 3, we observe that the rate of NE is about 3.7%, i.e. more than one NE every 30 words and about one NE every 2–3 sentences (with 10–15 tokens per sentence). Row 4 provides the number of words absent from our 200,000 item word list: this row gathers the NEs, the misspelled words and some rare unknown words. Let us note that we found a couple of NEs, which were listed in the list of well-formed tokens: this is for example the case for some very common proper names, which belong to the regular French vocabulary (as for example *Pierre*, which means *stone*).

Row 2 indicates the number of corrupted words produced by the corruption process. It serves to compute the successful correction rates given in row 5. Row 4 provides the result of a NER-free spelling corrector applied to the evaluation sample: we assume that such system would replace any unrecognized word, i.e. not only misspellings but also most named-entities.

In rows 5 and 6, results of the fully automatic spelling correction with pre-processing of NEs (columns ‘‘with NER’’) are directly interpretable in contrast with results provided without pre-processing of NEs (columns ‘‘Baseline’’). A first observation concerns the importance of the threshold, since it is probably better not to replace a misspelled item rather than replacing it by the wrong word, it is interested to limit the number of string operation of the string distance calculus. We call this property: *minimal commitment*, i.e. the ability of a system to evaluate whether it is better not to modify the data (see [12] for an extended example).

In row 5, results show that out of 361 corrupted words, 334 were appropriately corrected for Run 1 (92.5%), and 343 for Run 2 (95.0%). As expected, augmenting the threshold results in accepting more correction operations. Unfortunately, as side effect, it also results in some additional erroneous corrections (3.6% versus 2.6%, in row 6). More important, in row 6, we see that for each run the preprocessing of NEs does reduce the amount of erroneous correction. As expected, it means that whatever threshold value we select, preprocessing the document with a NER does reduce spelling correction errors: from more than 20% (21.8 and 23.5%) down to about 3% (2.6 and 3.6%).

5. Conclusion

We have shown that spelling correction can benefit from NLP tools such as named-entity recognizers and lexical disambiguation tools. It is also observed that part-of-speech taggers are more effective than word-sense taggers for broad-coverage purposes. These results confirm that NLP can provide an effective approach for labor-intensive tasks, which are until now conducted by healthcare professionals. We also showed evidences that NLP could radically transform some of these tasks into more automatic/fully automatic processes. Examples of such promising applications have been provided elsewhere: [9] for coding purposes, and [5] for clinical research.

In conclusion, we would like to suggest a future research direction for those concerned with processing clinical narratives. We believe that the tools and methods described in this article are mature enough—if not ready—to be imported in ‘‘real’’ biomedical information systems, and that the barriers are usually more cultural and educational than technical.

Thus, natural language processing could provide a synthetic answer in the current debate opposing structured data entries versus full-text data entries [19] in the electronic patient record (EPR), because information extraction based on NLP tools offers the possibility to transform full-text data entries into structured ones and vice versa (via generation). This position is clearly speculative, because it is not possible for current NLP systems to deliver such advanced features, but it remains that the NLP answer is synthetic in the way that is does not take position in traditional terms (like *reusability* and *maintainability* as opposed to *expressiveness* . . .) and instead aims at merging both antithetic alternatives in a common framework.

Acknowledgements

The study presented in this paper is supported by the Swiss National Foundation (MEDTAG Project, Grants 32-45’228.01 and 3200-065228). We would like to thank Berry De Bruijn, from the NRC (Ottawa, Canada) for helping with the style, the English language and the overall organization of the paper. Finally, we would like to thank Klaus-Peter Adlassnig and the referees for their helpful comments.

References

[1] Baud R, Lovis C, Ruch P, Rassinoux A-M. A light knowledge model for linguistic applications. In: Proceedings of AMIA; 2001. p. 37–41.

[2] Bikel D, Miller S, Schwartz R, Weischedel R. Nymble: a high-performance learning name-finder. In: Proceedings of ANLP. San Fransisco (CA): Morgan Kaufmann; 1997. p. 194–201.

[3] Bouillon P, Baud R, Robert G, Ruch P. Indexing by statistical tagging. In: Proceedings of the JADT’2000. Lausanne; 2000. p. 35–42.

[4] Brill E, Moore RC. An improved error model for noisy channel spelling correction. In: Proceedings of the ACL. San Fransisco (CA): Morgan Kaufmann; 2000. p. 286–93.

[5] de Bruijn LM, Hasman A, Arends JW. Automatic SNOMED classification—a corpus based method. Comp Programs Methods Biomed 1997;54:115–22.

[6] Church KW, Gale WA. Probability scoring for spelling correction. Stat Comp 1991;1:93–103.

[7] Courtin J, Dujardin D, Kowarski I, Genthial D, De Lima VL. Towards a complete detection/correction system. In: Proceedings of the ICCICL. Malaysia: Penang; 1991. p. 158–73.

[8] Damerau FJ. A technique for computer detection and correction of spelling errors. Commun ACM 1964;7(3):171–6.

[9] Franz P, Zaiss A, Schulz S, Hahn U, Klar R. Automated coding of diagnoses—three methods compared. In: Proceedings of AMIA; 2000. p. 250–4.

[10] Golding AR, Roth D. Applying Winnow to context-sensitive spelling correction. In: Proceedings of ICML. San Fransisco (CA): Morgan Kaufmann; 1996. p. 182–90.

[11] Golding AR, Shabes Y. Combining trigram-based and feature-based methods for context-sensitive spelling correction. In: Proceedings of the ACL. San Fransisco (CA): Morgan Kaufmann; 1997. p. 71–8.

[12] Hersh WR. Information retrieval at the millenium. In: Proceedings of the AMIA; 1998. p. 38–45.

[13] Hobbs J, Appelt D, Bear J, Israel D, Kameyama M, Stickel M, Tyson M. FASTUS: a cascaded finite-state transducer for extracting information from natural-language text. In: Finite State Devices for Natural Language Processing. Cambridge (MA): MIT Press; 1996.

[14] Jurafsky D, Martin JH. Speech and language processing. London: Prentice-Hall; 2000.

[15] Kantor P, Voorhees E. The TREC-5 confusion track: comparing retrieval methods for scanned text. Information Retrieval 2000;2:165–76.

[16] Klavans J, Resnik Ph. The Balancing Act combining symbolic and statistical approaches to language. Cambridge (MA): MIT Press; 1996.

[17] Lambert BL. Predicting look-alike and sound-alike medication errors. Am J Health Syst Pharm 1997;54:1161–71.

[18] Lilley LL, Guancy R. Sound-alike cephalosporins. How drugs with similar spellings and sounds can lead to serious errors. Am J Nut 1995;95(6):14–21.

[19] Lovis C, Baud RH, Revillard C. Paragraph oriented structure for narratives in medical documentation. In: Patel V, Rogers R, Haux R, editors. Proceedings of MEDINFO. Amsterdam: IOS Press; 2001. p. 638–42.

[20] Mangu L, Brill E. Automatic rule acquisition for spelling correction. In: Proceedings of ICML. San Fransisco (CA): Morgan Kaufmann; 1997. p. 187–94.

[21] Mays E, Damerau F, Mercer RL. Context based spelling correction. Information Processing Manage 1991;27(5):517–22.

[22] McCray A, Burgun A, Bodenreider O. Aggregating UMLS semantic types for reducing conceptual complexity. In: Patel V, Rogers R, Haux R, editors. Proceedings of MEDINFO. Amsterdam: IOS Press; 2001. p. 216–20.

[23] Mikheev A, Moens M, Grover C. Named entity recognition without gazetteers. In: Proceedings of EACL. San Fransisco (CA): Morgan Kaufmann; 1999. p. 47–55.

[24] MuchMore, State of the art report: http://www.muchmore.dfki.de/pub.html; 2001 [COIL rep.].

[25] National Library of Medicine. Documentation, UMLS Knowledge Sources, 12th ed. January 2001.

[26] Oflazer K. Error-tolerant finite state recognition with applications to morphological analysis and spelling correction. Comp Linguistics 1996;22(1):1–18.

[27] Peterson JL. Computer programs for detecting and correcting spelling errors. Comp Practices Commun ACM 1980;23(12):676–86.

[28] Pollock JJ, Zamora A. Automatic spelling correction in scientific and scholarly text. ACM Comp Surveys 1987;27(4):358–68.

[29] Reiner J, Gierl L. Case-based reasoning for antibiotics therapy advice: an investigation of retrieval algorithms and prototypes. Art Intel Med 2001;23:171–86.

[30] Ristad E, Yanilos P. Learning string edit distance. In: Proceedings of the ICML. San Fransisco (CA): Morgan Kaufmann; 1997. p. 522–32.

[31] Rivest RL. Learning decision lists. Machine Learning 1987;2:229–46.

[32] Ruch P. Using contextual spelling correction to improve retrieval effectiveness in degraded text collections. In: Proceedings of COLING 2002. San Fransisco (CA): Morgan Kaufman; 2002. p. 345–53.

[33] Ruch P, Baud R, Rassinoux A, Bouillon P, Robert G. Medical document anonymization with a semantic lexicon. In: Proceedings of AMIA; 2000. p. 729–33.

[34] Ruch P, Gaudinat A. Comparing corpora and lexical ambiguity. In: Kilgariff A, Berber Sardinha T, editors. Proceedings of the Workshop on Comparing Corpora. Hong-Kong; 2000. p. 14–9.

[35] Ruch P, Baud R, Bouillon P, Rassinoux A-M, Robert G. Tagging medical texts: a rule-based experiment, in medical lnfobahn for Europe. In: Hasman A, Blobel B, Dudeck J, Engelbrecht R, Gell G, Prokosch H-U, editors. Proceedings of MIE. Amsterdam: IOS Press; 2000. p. 448–55.

[36] Ruch P, Baud R, Bouillon P, Rassinoux A-M, Scherrer J-R. MEDTAG: tag-like semantics for medical document indexing. In: Proceedings of the AMIA; 1999. p. 137–41.

[37] Ruch P, Baud R, Bouillon P, Robert G. Minimal commitment and full lexical disambiguation: Balancing Rules and Hidden Markov Models. In: Proceedings of CoNLL (ACL-SIGNLL). San Fransisco (CA): Morgan Kaufmann; 2000. p. 111–5.

[38] Sweeney L. Replacing personally-identifying information in medical records, the scrub system. In: Cimino JJ, editor. Proceedings of the AMIA; 1996. p. 333–7.

[39] Yarowsky D. Decision lists for lexical ambiguity resolution: application to accent restoration in Spanish and French. In: Proceedings of ACL. San Fransisco (CA): Morgan Kaufmann; 1994. p. 88–95.

[40] Zeng Q, Kogan S, Ash N, Greenes RA. Patient and clinician vocabulary: how different are they? In: Patel V, Rogers R, Haux R, editors. Amsterdam: IOS Press; 2001. p. 399–403.