Reformatting Clinical Records Based on Global Layout Statistics

Pierre Zweigenbaum
LIMSI–CNRS
Rue John von Neumann
91405 Orsay, France
pz@limsi.fr

Cyril Grouin
LIMSI–CNRS
Rue John von Neumann
91405 Orsay, France
grouin@limsi.fr

Abstract

Texts in electronic medical records often keep traces of the layout used to display or print them in hospital information systems. This includes blank lines in double-spaced text or short lines in wrapped paragraphs. Directly inputting this format to an information extraction system may prevent it from optimally detecting entities or relations. We address this problem with a method of text normalization based on global statistics on the layout of a text. Our goal is to obtain reformatted texts in which sentences are never split across two lines. We present two evaluations: (i) intrinsic, which obtains very good results, with all double-spaced texts correctly detected, and wrapped line detection scores of $R = .9877$, $P = .9434$, $F = .9651$; (ii) extrinsic, which shows a small impact of text reformatting on two information extraction tasks: de-identification and risk factors identification (i2b2/UTHealth 2014 challenge). Precision is consistently improved at the expense of recall. The overall improvement on F-measure is small ($+.2$pt for PHI) or null (for risk factors). Closer examination for risk factors shows that it is negative ($-.2$pt F) for singled-spaced texts which incur wrapped lines, but positive ($+.7$pt F) for the larger set of double-spaced texts in our corpus.

1 Introduction

Documents from the biomedical domain (e.g., scientific papers, clinical records, etc.) contain useful information for clinicians and researchers. Those documents need automatic processing to make the most of the information they contain. Since a few years, an increasing number of shared tasks (BioNLP-ST, i2b2 NLP Challenge, DDI) proposed to process such kinds of documents to access and represent their content: identification of entities, of relationships between entities, of interactions between drugs, of events on the patient's timeline, etc.

In the biomedical community, an effort was made to produce and distribute rich corpora. The MIMIC II database (Saeed et al., 2011) includes documents of several genres (nursing notes, imaging reports, etc.). The de-identified clinical records of this database keep their original format (e.g., fixed-width column, single/double spacing, tables reproduced without column separator, headers or footers inserted in the middle of a paragraph, etc.). Those particularities often cause sentences to be broken into several segments: this may cause entities to be split across two sentences, preventing their recognition; or entities and their properties or relations to be separated instead of present in the same sentence, also impairing their detection. More generally, broken sentences impede the application of part-of-speech taggers or syntactic parsers which often bring precious information for semantic processing. A phase of text normalization should therefore benefit subsequent processing of such texts.

Text normalization has been an important issue in the text-to-speech domain. Nowadays, it constitutes a crucial step to process informal documents such as documents from the Internet. A few papers address these issues: (Zhu et al., 2007) considered the normalization task as classification task of each token in a text, considering different types of tokens (standard word, non-standard word, punctuation mark, space and line break). They performed several experiments based on SVM, CRF and a cascade of approaches to normalize informally entered text (SMS and forum posts) through a classification process (e.g., for a line break: preserve line break, replace line break by space, delete line break). Using a CRF model, they achieved high results depending on the task they processed: extra line break ($F = .9375$), extra space ($F = .9457$), sentence boundary ($F = .9918$). They also addressed extra punctuation marks, unnecessary tokens, and case restoration.

In this paper, we present the method of text normalization we designed to process a corpus of electronic medical records. Our goal is to obtain texts where sentences are never split.

We observed three main phenomena which interact to account for the original format of the texts: line spacing, line folding, and extra spaces within lines. We address the problem as two main decisions to make: whether a text uses double line spacing, and whether a text uses line folding. These decisions, which are based on global text statistics obtained from a training corpus, drive a deterministic reformatting process which replaces with spaces or removes selected line breaks, normalizing extra spaces in the process.

We present two evaluations of this method. The first is intrinsic and measures the recall, precision, and F-measure of line break removal in the task of reformatting broken sentences. The second is extrinsic and measures the impact of this reformatting on two information extraction tasks: the identification of PHI and risk factors for diabetic patients, as per the i2b2/UTHealth 2014 NLP challenge Tasks 1 and 2.

2 Material

The corpus is composed of 1,304 de-identified clinical records extracted from the MIMIC II database. This corpus was used as part of the 2014 i2b2/UTHealth NLP challenge, whose aim was to identify risk factors for diabetic patients automatically from clinical records (Stubbs et al., 2014).

A corpus analysis revealed the following four possible document structures, from most to least conform to a 'clean' version:

- double spacing, fixed-width columns, indenting and multi spaces;
- single spacing, fixed-width columns, indenting and multi spaces;
- single spacing, normalized sentences, indenting and multi spaces;
- single spacing, normalized sentences, no indenting nor multi spaces.

For the risk factor extraction task, gold standard annotations were provided for the following types: risk factors *CAD, family history of CAD, diabetes mellitus, hyperlipidemia, hypertension, obesity, smoking status*, as well as *medication* used to treat those disorders.

For each risk factor, additional information was provided: *time* to indicate whether the risk occurs before, during or after the document creation time (DCT), *indicator* to indicate how the information was mentioned in the text (e.g. for hyperlipidemia: "mention", "high cholesterol", and "high LDL"), *type* to indicate the medication class (among several classes: "aspirin", "diuretic", "statin", etc.).

The corpus is split into several parts as illustrated in Table 1. The Training/Train sub-corpus was used to gather statistics to tune the system, while the Training/Dev sub-corpus was used for its intrinsic evaluation. The Testing sub-corpus was used for the extrinsic (information extraction) evaluation.

Part	#doc
Corpus	1,304
Training	790
Train	390
Dev	131
Test	269
Testing	514
Double spacing and line wrapping	125
Single spacing and line wrapping	98
No need for reformatting	291

Table 1: Sub-corpora used in the present experiments

3 Methods

3.1 Text normalization

The basic operation we need to perform on a text is to decide what to do with each line break: keep it, replace it with a space (i.e., unwrap the current line), or remove it. A specificity of our method lies in its making this decision after an examination of the global layout of the text, then using this information to drive reformatting.

3.1.1 Acquiring global layout statistics

We observe two main text-level properties to obtain this global view:

1. Whether the text is double-spaced. Double spacing introduces blank lines every other line in a text, in such a way that it contains at least as many blank lines as non-blank lines.

2. Whether the text uses line wrapping. Typically, line wrapping is used to fold lines when they exceed a set maximum line width (henceforth the *column width*). A characteristic of such a text is that the length of its lines tends to be close to this column width.

We compute scores to assess each of these properties for a given text.

We tested variations on the number of blank lines in a text, line lengths, length of initial indentation, and used them as features for a text clustering algorithm (we used Weka's EM-based clusterer, which does not require to set an a priori number of clusters). We ran this clusterer on the Training/train subcorpus: it created between five and seven clusters depending on the input features. This helped us determine the features most associated to the global layout properties of the texts, among which we kept the following two.

For double spacing, we measure the ratio B of blank lines (= empty lines, or lines containing only spaces) over the total number of lines in the text as per formula (1), where L is the set of lines in the text.

$$B(L) = \frac{\mid l \in L; l \text{ is blank} \mid}{\mid L \mid} \tag{1}$$

For line wrapping, we measure statistics over line lengths. Ideally, the maximum line length in the whole text should be close to the column width. However, for some reason, much larger lines sometimes occur in the texts we examined, hence one cannot trust this maximum line length. A more reliable method consists in observing the line length average and standard deviation. Average line length should be more robust to noise than the single maximum. Wrapped lines should have a length which is close to average, so that a text which enforces line wrapping should have a small standard deviation. Indeed, if the column width is large, the standard deviation might be larger too, so a more reliable statistics should be obtained by dividing the standard deviation by the average: this is the *coefficient of variation CV* of the initial variable, here the length of each line of L, as per formula (2). Note that blank lines are ignored when computing CV: this way, this variable is not sensitive to the amount of 'vertical space' inserted in the text, only to the lengths of non-blank lines.

$$CV_{length}(L) = \frac{stddev_{\,l \in L}(length(l))}{avg_{l \in L}(length(l))} \tag{2}$$

We decide for each text whether it uses double spacing and whether it uses line wrapping according to a threshold tuned on the Training/train subcorpus. For double spacing, we ordered the documents of this sub-corpus in ascending order of B and determined the threshold B_t over which all texts were double-spaced. For line wrapping we did the same for CV_{length}, selecting a ceiling CV_c in the middle of the zone where texts evolved from heavily using line wrapping to slightly using line wrapping. Proceeding this way allowed us to focus our examination of input texts on selected zones instead of having to annotate the full sub-corpus, as a supervised learning algorithm would have required. (This can be seen as a manual implementation of active learning, where document scores were used to focus human annotation on their uncertainty zone.)

3.1.2 Text reformatting process

Final processing goes through three stages. (1) If a text is categorized as using double spacing, it can be reformatted by removing blank lines every other line. (2) Indentation is then reduced, taking into account indentation blocks. This helps the subsequent processing of wrapped lines if any. (3) If a text is categorized as using line wrapping, it is not reformatted in its entirety. Instead, local constraints are enforced to decide locally whether a line should be pasted to the following line. Some of these constraints rely on the global line length statistics. Local processing takes into account:

- Section titles, assumed to be made of a large enough sequence of uppercase words, or of capitalized words followed by a colon ':'; they start a new line;
- Numbered lists, with variant formats; context constraints are enforced on the presence of other numbered items before or after the current line; each item starts a new line;
- Tabulated lists; local homogeneity constraints are enforced by comparing the current indent size to the average of those of neighboring lines; each item starts a new line;
- A very short line (smaller than average length by less than one standard deviation) always ends the current reformatted line;
- A moderately short line (smaller than average length by less than half a standard deviation) ends the current reformatted line if it ends with a strong punctuation ('.', '!' or '?'); if it looks like a title (is are capitalized and contains a colon ':'), it makes a reformatted line by itself; otherwise it is treated as a normal line;
- Idiosyncrasies found in the Training/train corpus, such as initial identifiers (a coded line with uppercase letters and caret) and signature separation lines (made of underline characters) are kept as separate lines;
- Other lines are pasted to the preceding line.

Note that internally, tokens retain their original character offsets, thereby keeping their link to other existing annotations indexed on character offsets.

3.2 Application use case

3.2.1 Presentation

In order to test the impact of such automatic text normalization on NLP tasks, we applied a pipeline to process two tasks from the 2014 i2b2/UTHealth NLP Challenge (Stubbs et al., 2014). The first task aims at identifying PHI (protected health information) within clinical records among 7 main categories and 25 sub-categories. The second task consists in identifying 8 types of information from the same clinical records: diseases, associated risk factors for diabetic patients, and medication names. While outputs from the first task are expected to give offsets of characters for each found PHI, outputs from the second task must give information at the document level.

Our pipeline relies on two steps: (i) a tool to identify PHI or risk factors, based on Conditional Random Fields as implemented in the Wapiti tool (Lavergne et al., 2010), and (ii) a classifier (SVM) from the Weka toolkit (Hall et al., 2009) to process the "time" attribute associated with each risk factor. The details on this pipeline are presented in (Grouin, 2014; Grouin et al., 2014).

We use here this pipeline in two situations: (i) directly on the *original* texts of the corpus, and (ii) on the reformatted versions of these texts as obtained by the above-described process (*normalized*).

To obtain a finer-grained assessment of the differences, we split the Testing corpus (514 documents) into three sub-corpora (see Table 1):

- 125 documents for which double-spacing removal is needed, generally followed by reformatting wrapped lines ("double" sub-corpus hereafter);
- 98 documents with single spacing for which reformatting wrapped lines is needed ("single" sub-corpus hereafter);
- and the remaining 291 documents for which no reformatting is needed at all; although a small number of false positive reformatted lines cannot be fully excluded for this corpus, we estimate that their potential impact is very low and therefore that those documents are not relevant for our purpose.

This division into sub-corpora was based on a manual examination of the text representation produced by our pipeline.

3.2.2 Experiments

We produced two CRF models to identify risk factors:

- a model built on the original training corpus (790 documents for which we did not perform any text normalization, “original model”);
- and a model built on the training corpus with text normalization processing (“normalized model”).

We conducted four experiments:

1. the original model on the original single sub-corpus;
2. the original model on the original double sub-corpus;
3. the normalized model on the normalized single sub-corpus;
4. the normalized model on the normalized double sub-corpus.

3.3 Evaluation

3.3.1 Intrinsic evaluation

We examined the detection of double spacing at the level of each document in the Training/dev sub-corpus. 54% of the documents are double-spaced. We counted the proportion of these that were detected.

For line wrapping, we recorded the number of lines that were wrapped and should therefore be pasted to the next line (gold standard). We counted the number of wrapped lines that were correctly detected by our system (TP), the number of wrapped lines that were not detected by our system (FN), and the number of normal lines that were incorrectly pasted to the next line (FP). This allowed us to compute recall ($R = \frac{TP}{TP+FN}$), precision ($P = \frac{TP}{TP+FP}$), and F-measure ($F = \frac{2PR}{P+R}$).

3.3.2 Extrinsic evaluation

We used the i2b2/UTHealth evaluation script to compute the results of each information extraction experiment. On PHI, the evaluation script scores the following information for each annotation: (i) category of PHI (among the 7 main categories), (ii) “type” attribute (which corresponds to a potential sub-category among the 25 ones), (iii) starting and ending offsets, and (iv) identity of the contents of the “text” attribute and of the portion of text delimited by those offsets within the clinical record. Six evaluations in micro/macro averaged recall, precision and F-measure (Manning and Schütze, 2000) are provided; here we only report the strict evaluation on all categories of PHI, which is used for the official ranking.

On risk factors, the script scores complete annotations for each risk factor:

- “time” and “type” attributes for *medication*;
- “time” and “indicator” attributes for *CAD, diabetes, hyperlipidemia, hypertension, obese*;
- “status” attribute for *smoker*;
- “indicator” attribute for *family history*.

It computes micro-averaged recall, precision, and F-measure.

4 Results

The threshold B_t for the blank line ratio was set to 0.5: documents with $B \geq 0.5$ were submitted to removal of every other blank line.

The ceiling CV_c for the coefficient of variation of line length was set to 0.64: texts with $CV_{length} < 0.64$ were submitted to reformatting according to the process described in Section 3.1.2.

4.1 Intrinsic evaluation

All double-spaced documents were correctly detected and turned into single-spacing.

Table 2 shows the recall, precision and F-measure of line wrapping. 1,367 lines were correctly detected as being wrapped, 82 were incorrectly detected as wrapped, and 17 wrapped lines were not detected.

Sub-corpus	P	R	F
Training/dev	.9434	.9877	.9651

Table 2: Performance of wrapped line detection

4.2 Extrinsic evaluation

Table 3 shows the overall micro averaged results we achieved on both PHI and risk factor i2b2/UTHealth tasks on the test corpora, according to whether or not the texts have been reformatted. In both cases, precision is improved (+.8 or +.3pt) at the cost of a loss in recall (−.2pt), resulting in a slightly improved (PHI, +.2pt) or stable F-measure (risk factors, +0pt).

i2b2 task	Processing	P	R	F
PHI	Original	.8937	.7332	.8055
	Normalized	.9015	.7314	.8076
Risk factor	Original	.9057	.7922	.8451
	Normalized	.9085	.7903	.8453

Table 3: Micro averaged precision, recall and F-measure on PHI and risk factor identification i2b2/UTHealth tasks depending on whether or not the texts have been reformatted

Table 4 examines more closely the differences in the results of the risk factor identification pipeline achieved on the "single" and "double" sub-corpora, depending on whether the corpora used to build the CRF models and to evaluate the results contained the original texts or their normalized versions. The variation when moving from original to normalized texts ranges from −.5 percentage points (recall on single-spaced corpus) to +1 (precision on double-spaced corpus). Precision is always improved (+.2 or +1pt), while the orientation of recall depends on the sub-corpus (−.5 on Single, +.5 on Double) and drives the direction of variation of F-measure.

Processing	Sub-corpus	P	R	F
Original	Single	.8761	.7755	.8227
	Double	.8887	.8174	.8516
Normalized	Single	.8779	.7705	.8207
	Double	.8984	.8222	.8586

Table 4: Micro averaged precision, recall and F-measure on the risk factor identification task depending on the sub-corpus and whether or not the texts have been reformatted

5 Discussion

The intrinsic evaluations show that our text normalization system, tuned with statistics gathered from the Training/train sub-corpus, performs well on the Training/dev sub-corpus: it detects all double-spaced texts and detects wrapped lines with high recall and precision: its F-measure of .9651 is comparable to that of (Zhu et al., 2007) for extra line break detection (.9375). A large part of the lines incorrectly detected as wrapped occur in the header or footer parts of the reports: we expect that a specific classifier could detect these specific parts of the text and be used to prevent this spurious detection (see, e.g., (Deléger et al., 2014)).

The extrinsic evaluation obtains different results depending on the task. The detection of PHI is an entity detection task where each individual mention must be detected, together with its attributes. In contrast, the detection of risk factors as defined in the i2b2/UTHealth 2014 Task 2 can be seen as a text classification task: given a text, assert whether the patient had this or that risk factor (and when, relative to the date of the current document). In the latter task, whether one or multiple mentions of the same risk factor were present in the input text is not relevant. Therefore, missing one mention is not a problem if another mention of the same risk factor was present and correctly found in the text. This may explain why risk factor detection was less sensible to reformatting than PHI detection.

Looking more closely at the results for risk factor detection, we observed that it varied depending on the formatting properties of the considered sub-corpus. In the single-spaced sub-corpus (98 documents), precision increases by .2pt while recall decreases by .5pt, resulting in a lower F-measure (−.2pt). In contrast, in the double-spaced sub-corpus (125 documents), both precision (+1pt) and recall (+.5pt) increase, yielding an increase of F-measure (+.7pt). Reformatting has a larger, positive impact on texts which are most transformed by it: in the double-spaced sub-corpus, original texts with both double spacing and wrapped lines are seen as a series of small lines, many of which are only sub-sentential fragments. Once reformatted, sentences recover their integrity.

While the intrinsic evaluation is very positive, it actually results in modifying 1,449 lines in the Training/dev corpus out of a total of 19,007 lines, i.e., 7.6%. This relatively small proportion of the corpus lines may help explain the relatively low impact of reformatting on our information extraction tasks. Another explanation might be that in these tasks, few entities spanned two lines, or that our information extraction pipeline is robust to these phenomena. We examined further the distribution of intrinsic false positives in the Training/dev corpus and noticed that double-spaced documents have an average of 0.48 false positives while single-spaced documents have an average of 0.80 false positives: assuming that the Test and Training corpora have similar properties, this correlates with the decrease in recall in the extrinsic evaluation. The reason why there are more false positives in wrapped sentence detection in single-spaced documents remains to be found.

A direction to improve the detection of wrapped lines consists in including more local information in the classification of line breaks, after double spacing has been processed. This could be done by modelling the problem as a supervised classification task, where a space (including plain space or line break) must be classified as plain space or line break; features could include information on the previous and next word, among which their capitalization and their prevalence in first, last, or other position in a line, information on the current and neighboring line lengths, on top of the global statistics computed in the present work. This can be done with no additional annotation, either on noisy input with the original texts, or on cleaner input with texts that have been reformatted automatically with our system.

In the present information extraction experiments, contrarily to the other information extraction components, the temporal relation detection component was not retrained on the reformatted corpus: this is a limitation of these experiments. Besides, we expect that working on better-segmented sentences should also help design better features for the detection of temporal relations.

6 Conclusion

Observing global text layout statistics allowed us to design a method for the automatic detection of documents which use double spacing or line wrapping. This method detected all double spaced texts and obtained high recall and precision in the detection of individual wrapped lines ($F = .9651$). The obtained text reformatting consistently improved the precision of two clinical information extraction tasks, albeit by a small amount, while generally decreasing recall. Its overall impact on F-measure was positive (PHI detection, $+.2$pt) or null (risk factor detection); it was found to be stronger for risk factor detection ($+0.7$pt F) in the subset of double-spaced texts with or without wrapped lines.

Further investigation is needed to better understand the sources of negative impact (-0.2pt F) in single-spaced texts. We have outlined directions for improvement in the detection of individual wrapped lines through space classification. Another direction of improvement for the information extraction task consists in taking advantage of the better sentence segmentation to extend the set of relevant features that can be fed to its classifier: this is the topic of our current work.

Acknowledgments

This work was supported by the French National Agency for Research under grant Accordys[1] ANR-12-CORD-007-03.

References

Louise Deléger, Cyril Grouin, and Aurélie Névéol. 2014. Automatic content extraction for designing a French clinical corpus. In *AMIA Poster*, Washington, DC.

Cyril Grouin, Véronique Moriceau, Sophie Rosset, and Pierre Zweigenbaum. 2014. Risk factor identification from clinical records for diabetic patients. In *Proc of i2b2/UTHealth NLP Challenge*.

Cyril Grouin. 2014. Clinical records de-identification using CRF and rule-based approaches. In *Proc of i2b2/UTHealth NLP Challenge*.

Mark A. Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. 2009. The WEKA data mining software: An update. *SIGKDD Explor Newsl*, 11(1).

Thomas Lavergne, Olivier Cappé, and François Yvon. 2010. Practical very large scale CRFs. In *Proc of ACL*, pages 504–13, Uppsala, Sweden.

Christopher D. Manning and Hinrich Schütze. 2000. *Foundations of Statistical Natural Language Processing*. MIT Press, Cambridge, Massachusetts.

[1] Accordys: Agrégation de Contenus et de COnnaissances pour Raisonner à partir de cas de DYSmorphologie fœtale.

Mohammed Saeed, Mauricio Villaroel, Andrew T. Reisner, Gari Clifford, Li-Wei Lehman, George Moody, Thomas Heldt, Tin H. Kyaw, Benjamin Moody, and Roger G. Mark. 2011. Multiparameter Intelligent Monitoring in Intensive Care II (MIMIC-II): A public-access intensive care unit database. *Crit Care Med*, 39(5):952–60.

Amber Stubbs, Christopher Kotfila, Hua Xu, and Özlem Uzuner. 2014. Practical applications for NLP in clinical research: the 2014 i2b2/UTHealth shared tasks. In *Proc of i2b2/UTHealth NLP Challenge*.

Conghui Zhu, Jie Tang, Hang Li, Hwee Tou Ng, and Tiejun Zhao. 2007. A unified tagging approach to text normalization. In *Proc of ACL*, pages 688–95, Prague, Czech Republic.