# On the Effects of Low-Quality Training Data on Information Extraction from Clinical Reports

Diego Marcheggiani and Fabrizio Sebastiani

Istituto di Scienza e Tecnologia dell'Informazione
Consiglio Nazionale delle Ricerche
Via Giuseppe Moruzzi 1 – 56124 Pisa, Italy
{diego.marcheggiani,fabrizio.sebastiani}@isti.cnr.it

**Abstract.** In the last five years there has been a flurry of work on information extraction from clinical documents, i.e., on algorithms capable of extracting, from the informal and unstructured texts that are generated during everyday clinical practice, mentions of concepts relevant to such practice. Most of this literature is about methods based on supervised learning, i.e., methods for training an information extraction system from manually annotated examples. While a lot of work has been devoted to devising learning methods that generate more and more accurate information extractors, little work (if any) has been devoted to investigating the effect of the quality of training data on the learning process. Low quality in training data sometimes derives from the fact that the person who has annotated the data is different (e.g., more junior) from the one against whose judgment the automatically annotated data must be evaluated. In this paper we test the impact of such data quality issues on the accuracy of information extraction systems oriented to the clinical domain. We do this by comparing the accuracy deriving from training data annotated by the authoritative coder (i.e., the one who has annotated the test data), with the accuracy deriving from training data annotated by a different coder. The results indicate that, although the disagreement between the two coders (as measured on the training set) is substantial, the difference in accuracy is not so. This hints at the fact that current learning technology is robust to the use of training data of suboptimal quality.

## 1 Introduction

In the last five years there has been a flurry of work on information extraction from clinical documents, i.e., on algorithms capable of extracting, from the informal and unstructured texts that are generated during everyday clinical practice (e.g., admission reports, radiological reports, discharge summaries, clinical notes), mentions of concepts relevant to such practice. Most of this literature is about methods based on supervised learning, i.e., methods for training an information extraction system from manually annotated examples. The use of learning algorithms based on support vector machines (SVMs – [10, 14, 20]),

hidden Markov models (HMMs – [15]), and conditional random fields (CRFs – [4, 10, 11, 14, 16, 25, 27]) has been proposed; of these, CRFs have lately become the most popular, and have become the *de facto* standard in clinical information extraction.

While a lot of work has been devoted to devising text representation methods and variants of the aforementioned supervised learning methods that generate more and more accurate information extractors, little work (if any) has been devoted to investigating the effects of the quality of training data on the learning process. Issues of quality in the training data may arise for different reasons:

1. In several organizations it is often the case that the original annotation is performed by coders (a.k.a. "annotators", or "assessors") as a part of a daily routine in which fast turnaround, rather than annotation quality, is the main goal of the coders and/or of the organization. An example is the (increasingly frequent) case in which annotation is performed via crowdsourcing (e.g., Mechanical Turk, CrowdFlower, etc.[1]) [7, 21].
2. In many organizations it is also the case that annotation work is usually carried out by junior staff (e.g., interns), since having it accomplished by senior employees would make costs soar.
3. It is often the case that the coders entrusted with the annotation work were not originally involved in designing the tagset (i.e., the set of concepts whose mentions are sought in the documents). As a result, the coders may have an imperfect understanding of the true meaning of these concepts, or of how their mentions are meant to look like, which may negatively affect the quality of their annotation.
4. The data used for training the system may sometimes be old or outdated, with the annotations not reflecting the current meaning of the concepts anymore. This is an example of a phenomenon, called *concept drift* [17, 19], which is well known in machine learning.

We may summarize all the cases mentioned above by saying that, should the training data be re-annotated by an *authoritative coder* (hereafter indicated as $C_\alpha$), the resulting annotations would be, to a certain extent, different. We would also be able to precisely measure this difference, by measuring the *intercoder agreement* (via measures such as Cohen's kappa – see e.g., [1, 3]) between the training data $Tr$ as coded by $C_\alpha$ and the training data as coded by whoever else originally annotated them (whom we will call, for simplicity, the *alternative coder* – hereafter indicated as $C_\beta$). In the rest of this paper we will take the authoritative coder $C_\alpha$ to be *the coder whose annotations are to be taken at face value*, i.e., considered as the "gold standard". As a consequence we may assume that $C_\alpha$ is the coder who, once the system is trained and applied, has also the authority to evaluate the accuracy of the automatic annotation (i.e., decide which annotations are correct and which are not)[2]. In this case, intercoder

[1] `https://www.mturk.com/`, `http://crowdflower.com/`

[2] For some organizations this authoritative coder may well be a fictional entity, e.g., several coders may be equally experienced and thus equally authoritative. However, without loss of generality we will hereafter assume that $C_\alpha$ exists and is unique.

disagreement measures the amount of noise that is introduced in the training data by having them annotated by a coder $C_\beta$ different from the authoritative coder $C_\alpha$.

It is natural to expect the accuracy of an information extraction system to be lower if the training data have been annotated by an alternative coder $C_\beta$, and higher if they have been annotated by $C_\alpha$ herself. However, note that this is *not* a consequence of the fact that $C_\alpha$ is more experienced, or senior, or reliable, than $C_\beta$. Rather, it is a consequence of the fact that standard supervised learning algorithms are based on the assumption that the training set and the test set are identically and independently distributed, i.e., that both sets are randomly drawn from the *same* distribution. As a result, these algorithms learn to replicate the subjective annotation style of their supervisors, i.e., of those who have annotated the training data. This means that we may expect accuracy to be higher simply when the coder of the training set and the coder of the test set are the *same* person, and to be lower when the two coders are different, irrespective of how experienced, or senior, or reliable, they are. In other words, the very fact that a coder is entrusted with the task of evaluating the automatic annotations (i.e., of annotating the test set) makes this coder *authoritative by definition*. For this reason, the authoritative coder $C_\alpha$ may equivalently be defined "the coder who has annotated the test set" (or: "the coder whose judgments we adhere to when evaluating the accuracy of the system"), and "alternative coder" to mean "a coder different from the authoritative coder".

The above arguments point to the fact that the impact of training data quality (under its many facets discussed in items (1)-(4) above) on the accuracy of information extraction systems may be measured by

1. evaluating the accuracy of the system in a *homogeneous* setting (i.e., both training and test sets annotated by the authoritative coder $C_\alpha$), and then
2. evaluating the loss in accuracy, with respect to the homogeneous setting, that derives from working instead in a *heterogeneous* setting (i.e., test set annotated by $C_\alpha$ and training set annotated by an alternative coder $C_\beta$)[3].

In this paper we test the impact of training data quality on the accuracy of information extraction systems oriented to the clinical domain. We do this by testing the accuracy of a state-of the-art, CRFs-based system on a dataset of radiology reports (originally discussed in [4]) in which a portion of the data has independently been annotated by two different experts. In other words, we try to answer the question: "What is the consequence of the fact that my training data are not sterling quality? that the coders who produced them are not entirely

[3] In the domain of classification the homogeneous and heterogeneous settings have also been called *self-classification* and *cross-classification*, respectively [28]. We depart from this terminology in order to avoid any confusion with *self-learning* (which refers to retraining a classifier by using, as additional training examples, examples the classifier itself has classified) and *cross-lingual classification* (which denotes a variant of text classification which exploits synergies between training data expressed in different languages).

dependable? How much am I going to lose in terms of accuracy of the trained system?"

In these experiments we not only test the "pure" homogeneous and heterogeneous settings described above, but we also test *partially heterogeneous* settings, in which increasingly large portions of the training data as annotated by $C_\alpha$ are replaced with the corresponding portions as annotated by $C_\beta$, thus simulating the presence of incrementally higher amounts of noise. For each setting we compute the intercoder agreement between the two training sets; this allows us to study the relative loss in extraction accuracy as a function of the agreement between authoritative and alternative assessor as measured on the training set. Since in many practical situations it is easy to compute (or estimate) the intercoder disagreement between (a) the coder to whom we would ideally entrust the annotation task (e.g., a senior expert in the organization), and (b) the coder to whom we can indeed entrust it given time and cost constraints (e.g., a junior member of staff), this will give the reader a sense of how much intercoder disagreement generates how much loss in extraction accuracy.

The rest of the paper is organized as follows. Section 2 reviews related work in information extraction from clinical documents and on establishing the relations between training data quality and extraction accuracy. In Sections 3 and 4 we describe experiments that attempt to quantify the degradation in extraction accuracy that derives from low-quality training data, with Section 3 devoted to spelling out the experimental setting and Section 4 devoted instead to presenting and discussing the results. Section 5 concludes, discussing avenues for further research.

## 2 Related Work

The literature on the effects of imperfect training data quality on prediction accuracy is extremely scarce, even within the machine learning literature at large. An early such study is [18], who look at these issues in the context of learning to predict prices of mutual funds from economic indicators. Differently from us, the authors work with noise artificially inserted in the training set, and not with naturally occurring noise. From experiments run with a linear regression model they reach the bizarre conclusion that "the predictive accuracy (...) is better when errors exist in training data than when training data are free of errors.", while the opposite conclusion is (somehow more expectedly) reached from experiments run with a neural networks model. A similar study, where the context is predicting the average air temperature in distributed heating systems, was carried out in [9]; yet another, where the goal was predicting the production levels of palm oil via a neural network, is [12].

In the context of a biomedical information extraction task[4] Haddow and Alex [8] examine the situation in which training data annotated by two different

[4] Biomedical IE is different from clinical IE, in that the latter (unlike the former) is usually characterized by idiosyncratic abbreviations, ungrammatical sentences, and sloppy language in general. See [6, Section 5] for a discussion of this point.

coders are available, and they found that higher accuracy is obtained by using both versions at the same time than by attempting to reconcile them or using just one of them. Their use case is different from ours, since in the case we discuss only one set of annotations, those of the alternative coder, are available as training data. Note also that training data annotated by more than one coder are rarely available in practice.

Closer to our application context, Esuli and Sebastiani [6] have thoroughly studied the effect of imperfect training data quality in text classification. However, in their case the degradation in the quality of the training data is obtained, for mere experimental purposes, via the insertion of artificial noise, due to the fact that their datasets did not contain data annotated by more than one coder. As a result, it is not clear how well the type of noise they introduce models naturally occurring noise. Webber and Pickens [28] also address the text classification task (in the context of e-discovery from legal texts), bur differently from [6] they work with naturally occurring noise. Differently from the present work, the multiply-coded training data they use were coded by one coder known to be an expert coder and another coder known to be a junior coder; our work instead (a) focuses on information extraction, and (2) does not make any assumption on the relative level of expertise of the two coders.

## 3 Experimental setting

### 3.1 Basic notation and terminology

Let us fix some basic notation and terminology. Let $\mathbf{X}$ be a set of texts, where we view each text $\mathbf{x} \in \mathbf{X}$ as a sequence $\mathbf{x} = \langle x_1, \ldots, x_{|\mathbf{x}|} \rangle$ of *tokens* (i.e., word occurrences) such that $x_{t_1}$ occurs before $x_{t_2}$ in the text (noted $x_{t_1} \preceq x_{t_2}$) if and only if $t_1 \leq t_2$. Any sequence of characters that separates two tokens in the text (e.g., a comma followed by a space) is called a *separator*. As a result, $|\mathbf{x}|$ denotes the length of the text. Let $C = \{c_1, \ldots, c_m\}$ be a predefined set of *concepts* (a.k.a. *tags*, or *markables*), or *tagset*. We take *information extraction* (IE) to be the task of determining, for each $\mathbf{x} \in \mathbf{X}$ and for each $c_r \in C$, a sequence $\mathbf{y}_r = \langle y_{r1}, \ldots, y_{r|\mathbf{x}|} \rangle$ of labels $y_{rt} \in \{c_r, \overline{c}_r\}$, which indicates which tokens in the text are labelled with tag $c_r$ and which are not. Since each $c_r \in C$ is dealt with independently of the other concepts in $C$, we hereafter drop the $r$ subscript and, without loss of generality, treat IE as the *binary* task of determining, given text $\mathbf{x}$ and concept $c$, a sequence $\mathbf{y} = \langle y_1, \ldots, y_{|\mathbf{x}|} \rangle$ of labels $y_t \in \{c, \overline{c}\}$.

Tokens labelled with a concept $c$ usually come in coherent sequences, or "segments". Hereafter, a *segment* $\sigma$ of text $\mathbf{x}$ for concept $c$ will be a pair $(x_{t_1}, x_{t_2})$ consisting of a start token $x_{t_1}$ and an end token $x_{t_2}$ such that (i) $x_{t_1} \preceq x_{t_2}$, (ii) all tokens $x_{t_1} \preceq x_t \preceq x_{t_2}$ are labelled with concept $c$, and (iii) the token that immediately precedes $x_{t_1}$ and the one that immediately follows $x_{t_2}$ are *not* labelled with concept $c$. In general, a text $\mathbf{x}$ may contain zero, one, or several segments for concept $c$.

### 3.2 Dataset

The dataset we have used to test the ideas discussed in the previous sections is the UmbertoI(RadRep) dataset first presented in [4], consisting of a set of 500 free-text mammography reports written (in Italian) by medical personnel of the Istituto di Radiologia of Policlinico Umberto I, Roma, IT. The dataset is annotated according to 9 concepts relevant to the field of radiology and mammography: BIR ("Outcome of the BIRADS test"), ITE ("Technical Info"), IES ("Indications obtained from the Exam"), TFU ("Followup Therapies"), DEE ("Description of Enhancement"), PAE ("Presence/Absence of Enhancements"), ECH ("Outcomes of Surgery"), DEP ("Prosthesis Description"), and LLO ("Locoregional Lymph Nodes"). Mentions of these concepts are present in the reports according to fairly irregular patterns. In particular, a given concept (a) need not be instantiated in all reports, and (b) may be instantiated more than once (i.e., by more than one segment) in the same report. Segments instantiating different concepts may overlap, and the order of presentation of the different concepts varies across the reports. On average, there are 0.87 segments for each concept in a given report, and the average segment length is 17.33 words.

The reports were annotated by two equally expert radiologists, Coder1 and Coder2; 191 reports were annotated by Coder1 only, 190 reports were annotated by Coder2 only, and 119 reports were annotated independently by Coder1 and Coder2. From now on we will call these sets `1-only`, `2-only` and `Both`, respectively; `Both(1)` will identify the `Both` set as annotated by Coder1, and `Both(2)` will identify the `Both` set as annotated by Coder2. The annotation activity was preceded by an alignment phase, in which Coder1 and Coder2 jointly annotated 20 reports (not included in this dataset) in order to align their understanding of the meaning of the concepts. See [4, Section 4.2] for a more detailed description of the UmbertoI(RadRep) dataset that includes per-concepts stats and other details[5].

### 3.3 Learning algorithm

As a learning algorithm we have used *linear-chain conditional random fields* (LC-CRFs - [13, 22, 23]), in Charles Sutton's GRMM implementation[6]. LC-CRFs is a class of supervised learning algorithms explicitly devised for *sequence labelling*, i.e., for learning to label items that naturally occur in sequences and such that the label of an item may depend on the features and/or on the labels of other items that precede or follow it in the sequence (which is indeed the case for the tokens in a text). LC-CRFs are members of the class of *graphical models*, a family of probability distributions that factorize according to an underlying graph [26]; see [23] for a full mathematical explanation of LC-CRFs.

A CRFs-based learner needs each token $x_t$ to be represented by a vector $\mathbf{x}_t$ of features. In this work we have used a set of features which includes one feature representing the word of which the token is an instance, one feature representing

[5] No other dataset is used in this paper since we were not able to locate another dataset of annotated clinical texts that contains reports annotated by more than one coder and is at the same time publicly available.

[6] `http://mallet.cs.umass.edu/grmm/`

its stem, one feature representing its part of speech, eight features representing its prefixes and suffixes (the first and the last $n$ characters of the token, with $n = 1, 2, 3, 4$), one feature representing information on token capitalization (i.e., whether the token is all uppercase, all lowercase, first letter uppercase, or mixed case), and 4 "positional" features [4, Section 3.3] that indicate in which half, 3rd, 4th, or 5th, respectively, of the text the token occurs in.

### 3.4 Evaluation measures

**Classification accuracy** As a measure of classification accuracy we use, as recommended in [4], the token-level variant (proposed in [5]) of the well-known $F_1$ measure, according to which an information extraction system is evaluated on an event space consisting of all tokens in the text. In other words, each token $x_t$ (rather than each segment, as in the traditional "segmentation F-score" model [24]) counts as a true positive, true negative, false positive, or false negative for a given concept $c_r$, depending on whether $x_t$ belongs to $c_r$ or not in the predicted annotation and in the true annotation. This model has the advantage that it credits a system for partial success (i.e., non-null overlap between a predicted segment and a true segment for the same concept), and that it penalizes both overannotation and underannotation.

As is well-known, $F_1$ combines the contributions of *precision* ($\pi$) and *recall* ($\rho$), and is defined as

$$F_1 = \frac{2\pi\rho}{\pi + \rho} = \frac{2TP}{2TP + FP + FN} \tag{1}$$

where $TP$, $FP$, and $FN$ stand for the numbers of true positives, false positives, and false negatives, respectively. Note that $F_1$ is undefined when $TP = FP = FN = 0$; in this case we take $F_1$ to equal 1, since the system has correctly annotated all tokens as negative.

We compute $F_1$ across the entire test set, i.e., we generate a single contingency table by putting together all tokens in the test set, irrespectively of the text they belong to. We then compute both *microaveraged* $F_1$ (denoted by $F_1^{\mu}$) and *macroaveraged* $F_1$ ($F_1^{M}$). $F_1^{\mu}$ is obtained by (i) computing the concept-specific values $TP_r$, $FP_r$ and $FN_r$, (ii) obtaining $TP$ as the sum of the $TP_r$'s (same for $FP$ and $FN$), and then (iii) applying Equation 1. $F_1^{M}$ is obtained by first computing the concept-specific $F_1$ values and then averaging them across the $c_r$'s.

**Intercoder agreement** As a measure of intercoder agreement we use Cohen's kappa (noted $\kappa$), defined as $\kappa = \frac{P(A)-P(E)}{1-P(E)}$, where $P(A)$ denotes the probability (i.e., relative frequency) of agreement and $P(E)$ denotes the probability of chance agreement (see [1, 3] for details). We opt for kappa since it is the most widely known, and best understood, measure of ICA. For Cohen's kappa too we work at the token level, i.e., for each token $x_t$ we record whether the two coders agree on whether $x_t$ is labeled or not with the concept $c$ of interest.

Incidentally, note that (as observed in [5]) we can compute Cohen's kappa only thanks to the fact that (as discussed in Section 3.4) we conduct our evaluation at the token level. Those who conduct their evaluation at the segment level (e.g., [2]) find that they are unable to do so, since in order to be defined kappa needs the notion of a true negative to be also defined, and this is undefined at the segment level. Evaluation at the segment level thus prevents the use of kappa and leaves $F_1$ as the only choice.

## 4 Results

### 4.1 Experimental protocol

In [4], experiments on the UmbertoI(RadRep) dataset were run using either `1-only` and/or `2-only` (i.e., the portions of the data that only one coder had annotated) as training data and `Both(1)` and/or `Both(2)` (i.e., the portion of the data that both coders had annotated, in both versions) as test data.

In this paper we switch the roles of training set and test set, i.e., use `Both(1)` or `Both(2)` as training set (since for the purpose of this paper we need training data with multiple, alternative annotations) and `1-only` or `2-only` as test set. Specifically, we run two batches of experiments: in Batch 1 Coder1 plays the role of the authoritative coder ($C_\alpha$) and Coder2 plays the role of the alternative coder ($C_\beta$), while in Batch 2 Coder2 plays the role of $C_\alpha$ and Coder1 plays the role of $C_\beta$. Each of the two batches of experiments is composed of:

1. An experiment using the homogeneous setting, i.e., both training and test data are annotated by $C_\alpha$. This means training on `Both(1)` and testing on `1-only` (Batch 1) and training on `Both(2)` and testing on `2-only` (2nd batch).
2. An experiment using the heterogeneous setting, i.e., training data annotated by $C_\beta$ and test data annotated by $C_\alpha$. This means training on `Both(2)` and testing on `1-only` (Batch 1) and training on `Both(1)` and testing on `2-only` (Batch 2).
3. Experiments using the partially heterogeneous setting, i.e., test data annotated by $C_\alpha$, and training data annotated in part by $C_\beta$ ($x$% of the training documents, chosen at random) and in part by $C_\alpha$ (the remaining $(100-x)$% of the training documents). We call $x$ the *corruption ratio* of the training set; $x = 0$ obviously corresponds to the fully homogeneous setting while $x = 100$ corresponds to the fully heterogeneous setting. We run experiments for each $x \in \{10, 20, ..., 80, 90\}$ by monotonically adding, for increasing values of $x$, new randomly chosen elements (10% at a time) to the set of training documents annotated by $C_\beta$. Since the choice of training data annotated by $C_\beta$ is random, we repeat the experiment 10 times for each value of $x \in \{10, 20, ..., 80, 90\}$, each time with a different random such choice.

For each of the above train-and test experiment we compute the intercoder agreement $\kappa(Tr, \widetilde{Tr})$ between the non-corrupted version of the training set $Tr$ and the corrupted version $\widetilde{Tr}$. We then take the average among the 10 values of

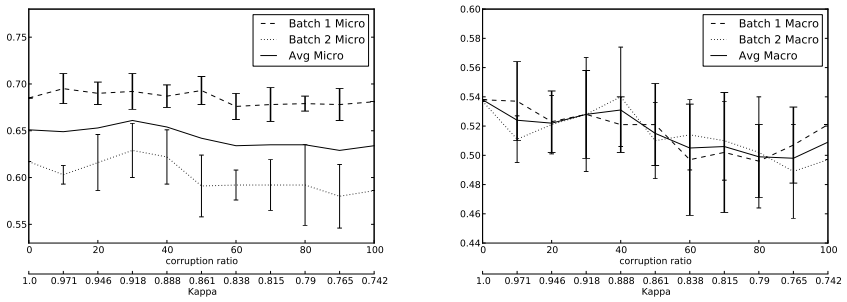


**Fig. 1.** Microaveraged $F_1$ (left) and macroaveraged $F_1$ (right) as a function of the fraction $x$ of the training set that is annotated by $C_\beta$ instead of $C_\alpha$ ("corruption ratio"). The dashed line represents the experiments in Batch 1, the dotted line represents those in Batch 2, and the solid one represents the average between the two batches. The vertical bars indicate, for each $x \in \{10, 20, ..., 80, 90\}$, the standard deviation across the 10 runs deriving from the 10 random choices of the elements in $Tr_\beta$.

$\kappa(Tr_\alpha, \widetilde{Tr}_\alpha)$ deriving from the 10 different experiments run for a given value of $x$ and denote it as $\kappa(x)$; this value indicates the average intercoder agreement that derives by "corrupting" $x$% of the documents in the training set, i.e., by using for them the annotations performed by the alternative coder.

For each of the above train-and test experiment we also compute the extraction accuracy (via both $F_1^\mu$ and $F_1^M$) and the relative loss in extraction accuracy that results from the given corruption ratio.

### 4.2 Results and discussion

Figure 1 illustrates the results of our experiments by plotting $F_1$ as a function of the corruption ratio $x$. For each value of $x$, the corresponding level of interannotator agreement $\kappa(x)$ (as averaged across the two batches) is also indicated. Table 1 reports precise extraction accuracy figures for the homogeneous and heterogeneous settings, for both batches of experiments and along with the resulting intercoder agreement values.

A first fact that emerges is that, for both $F_1^\mu$ and $F_1^M$, there is (as could be expected, and notwithstanding some oscillations) a clear decreasing pattern for accuracy as a function of $x$.

However, the most important (and perhaps surprising) fact is that this decrease is not dramatic. As specified in Table 1, a decrease in kappa values from 1.000 to 0.742 (a quite substantive decrease) determines a decrease in $F_1^M$ (average across the two batches) from 0.538 to 0.501 (-6.87%) and from 0.651 to 0.634 (-2.61%). In order to check whether this decrease is statistically significant we have performed a Mann-Whitney-Wilcoxon test (also known as the "Wilcoxon rank-sum test"), which determines if there is a statistically significant difference

**Table 1.** Extraction accuracy for the homogeneous setting ($x = 0$) and heterogeneous setting ($x = 100$), for both batches of experiments (and for the average across the two batches), and along with the resulting intercoder agreement values expressed as $\kappa(x)$.

| | $x$ | $\kappa(x)$ | $F_1^\mu$ | $F_1^M$ |
|---|---|---|---|---|
| Batch 1 | 0 | 1.00 | 0.685 | 0.538 |
| | 100 | 0.742 | 0.681 | 0.521 |
| Batch 2 | 0 | 1.00 | 0.617 | 0.537 |
| | 100 | 0.742 | 0.586 | 0.497 |
| Average | 0 | 1.00 | 0.651 | 0.538 |
| | 100 | 0.742 | 0.634 | 0.501 |

between the data automatically annotated at $x = 0$ corruption ratio and those annotated at $x = 100$ corruption ratio. In this test the difference is considered statistically significant if the resulting $p$ value is $< 0.05$. One advantage of this test is that, unlike the t-test, it does not require the data to be normally distributed. From this test we obtained a value of $p = 0.895$ for Batch 1 and a value of $p = 0.659$ for Batch 2; this clearly shows that the decrease in accuracy that we have suffered from by using training data labelled by an alternative coder is not statistically significant.

This seems to indicate that current learning technology is robust to the use of training data of suboptimal quality. The fact that the decrease is more marked for $F_1^M$ than for $F_1^\mu$ suggests that the most frequent codes are less affected than the most infrequent ones; this could be expected, since the most frequent codes have more training data, which somehow compensates for the suboptimal quality of these data.

This comparatively small decrease in accuracy might seem somehow surprising in the light of the results of [6], who show that, in a text classification context, minimal quantities of training noise can give rise to extremely substantial losses in classification accuracy. However, the key difference is that [6] uses artificially generated noise (since the datasets the authors use do not contain multiply annotated documents), and it is not clear that the noise model the authors use well represents the kind of noise we want to deal with here.

## 5 Conclusions

Few researchers, if any, have investigated the loss in accuracy that occurs when a supervised learning algorithm is fed with training data of suboptimal quality. We do this in the case of information extraction systems (trained via supervised learning) as applied to the detection of mentions of concepts of interest in medical notes. Specifically, we test to what extent extraction accuracy suffers from the fact that the person who has annotated the test data (the "authoritative coder"), who is by definition the person to whose judgment we conform irrespectively of her level of expertise, is different from the person who has labelled the training

data (the "alternative coder"). Our experimental results, that we have obtained on a dataset of 500 mammography reports annotated according to 9 concepts of interest, are somehow surprising, in the sense that they indicate that only a marginal decrease in extraction accuracy follows from a substantial decrease in training data quality. This seems to indicate that current supervised learning technology (and, in particular, the conditional random fields technology that we have used here) is robust to the use of training data of suboptimal quality. Since labelling cost is an important issue in the generation of training data (with senior coders costing much more than junior ones, and with internal coders costing much more than "mechanical turkers"), this result may give important indications as to the cost-effectiveness of low-cost annotation work.

## References


1. R. Artstein and M. Poesio. Inter-coder agreement for computational linguistics. *Computational Linguistics*, 34(4):555–596, 2008.
2. W. W. Chapman and J. N. Dowling. Inductive creation of an annotation schema for manually indexing clinical conditions from emergency department reports. *Journal of Biomedical Informatics*, 39(2):196–208, 2006.
3. B. Di Eugenio and M. Glass. The kappa statistic: A second look. *Computational Linguistics*, 30(1):95–101, 2004.
4. A. Esuli, D. Marcheggiani, and F. Sebastiani. An enhanced CRFs-based system for information extraction from radiology reports. *Journal of Biomedical Informatics*, 46(3):425–435, 2013.
5. A. Esuli and F. Sebastiani. Evaluating information extraction. In *Proceedings of the Conference on Multilingual and Multimodal Information Access Evaluation (CLEF 2010)*, pages 100–111, Padova, IT, 2010.
6. A. Esuli and F. Sebastiani. Training data cleaning for text classification. *ACM Transactions on Information Systems*, 31(4), 2013.
7. C. Grady and M. Lease. Crowdsourcing document relevance assessment with Mechanical Turk. In *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, pages 172—179, Los Angeles, US, 2010.
8. B. Haddow and B. Alex. Exploiting multiply annotated corpora in biomedical information extraction tasks. In *Proceedings of the 6th Conference on Language Resources and Evaluation (LREC 2008)*, Marrakech, MA, 2008.
9. S. Jassar, Z. Liao, and L. Zhao. Impact of data quality on predictive accuracy of ANFIS-based soft sensor models. In *Proceedings of the 2009 IEEE World Congress on Engineering and Computer Science (WCECS 2009)*, volume II, San Francisco, US, 2009.
10. M. Jiang, Y. Chen, M. Liu, S. T. Rosenbloom, S. Mani, J. C. Denny, and H. Xu. A study of machine-learning-based approaches to extract clinical entities and their assertions from discharge summaries. *Journal of the American Medical Informatics Association*, 18(5):601–606, 2011.
11. S. Jonnalagadda, T. Cohen, S. Wu, and G. Gonzalez. Enhancing clinical concept extraction with distributional semantics. *Journal of Biomedical Informatics*, 45(1):129—140, 2012.
12. A. Khamis, Z. Ismail, K. Haron, and A. T. Mohammed. The effects of outliers data on neural network performance. *Journal of Applied Sciences*, 5(8):1394–1398, 2005.

13. J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the 18th International Conference on Machine Learning (ICML 2001)*, pages 282–289, Williamstown, US, 2001.
14. D. Li, K. Kipper-Schuler, and G. Savova. Conditional random fields and support vector machines for disorder named entity recognition in clinical texts. In *Proceedings of the ACL Workshop on Current Trends in Biomedical Natural Language Processing (BioNLP 2008)*, pages 94–95, Columbus, US, 2008.
15. Y. Li, S. Lipsky Gorman, and N. Elhadad. Section classification in clinical notes using supervised hidden Markov model. In *Proceedings of the 2nd ACM International Health Informatics Symposium*, pages 744–750, Arlington, US, 2010.
16. J. Patrick and M. Li. High accuracy information extraction of medication information from clinical notes: 2009 i2b2 medication extraction challenge. *Journal of the American Medical Informatics Association*, 17:524–527, 2010.
17. J. Quiñonero-Candela, M. Sugiyama, A. Schwaighofer, and N. D. Lawrence, editors. *Dataset shift in machine learning*. The MIT Press, Cambridge, US, 2009.
18. D. F. Rossin and B. D. Klein. Data errors in neural network and linear regression models: An experimental comparison. *Data Quality Journal*, 5(1), 1999.
19. C. Sammut and M. Harries. Concept drift. In C. Sammut and G. I. Webb, editors, *Encyclopedia of Machine Learning*, pages 202–205. Springer, Heidelberg, DE, 2011.
20. T. Sibanda, T. He, P. Szolovits, and Ö. Uzuner. Syntactically-informed semantic category recognition in discharge summaries. In *Proceedings of the Annual Symposium of the American Medical Informatics Association (AMIA 2006)*, pages 714–718, Washington, US, 2006.
21. R. Snow, B. O'Connor, D. Jurafsky, and A. Y. Ng. Cheap and fast - but is it good? Evaluating non-expert annotations for natural language tasks. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing (EMNLP 2008)*, pages 254–263, Honolulu, US, 2008.
22. C. Sutton and A. McCallum. An introduction to conditional random fields for relational learning. In L. Getoor and B. Taskar, editors, *Introduction to Statistical Relational Learning*, pages 93–127. The MIT Press, Cambridge, US, 2007.
23. C. Sutton and A. McCallum. An introduction to conditional random fields. *Foundations and Trends in Machine Learning*, 4(4):267–373, 2012.
24. J. Suzuki, E. McDermott, and H. Isozaki. Training conditional random fields with multivariate evaluation measures. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the ACL (ACL/COLING 2006)*, pages 217—224, Sydney, AU, 2006.
25. M. Torii, K. Wagholikar, and H. Liu. Using machine learning for concept extraction on clinical documents from multiple data sources. *Journal of the American Medical Informatics Association*, 18(5):580–587, 2011.
26. M. J. Wainwright and M. I. Jordan. Graphical models, exponential families, and variational inference. *Foundations and Trends in Machine Learning*, 1(1/2):1–305, 2008.
27. Y. Wang and J. Patrick. Cascading classifiers for named entity recognition in clinical notes. In *Proceedings of the RANLP 2009 Workshop on Biomedical Information Extraction*, pages 42–49, Borovets, BG, 2009.
28. W. Webber and J. Pickens. Assessor disagreement and text classifier accuracy. In *Proceedings of the 36th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2013)*, pages 929–932. Dublin, IE, 2013.