Chunking Clinical Text Containing Non-Canonical Language

Aleksandar Savkov
Department of Informatics
University of Sussex
Brighton, UK
a.savkov@sussex.ac.uk

John Carroll
Department of Informatics
University of Sussex
Brighton, UK
j.a.carroll@sussex.ac.uk

Jackie Cassell
Primary Care and Public Health
Brighton and Sussex Medical School
Brighton, UK
j.cassell@bsms.ac.uk

Abstract

Free text notes typed by primary care physicians during patient consultations typically contain highly non-canonical language. Shallow syntactic analysis of free text notes can help to reveal valuable information for the study of disease and treatment. We present an exploratory study into chunking such text using off-the-shelf language processing tools and pre-trained statistical models. We evaluate chunking accuracy with respect to part-of-speech tagging quality, choice of chunk representation, and breadth of context features. Our results indicate that narrow context feature windows give the best results, but that chunk representation and minor differences in tagging quality do not have a significant impact on chunking accuracy.

1 Introduction

Clinical text contains rich, detailed information of great potential use to scientists and health service researchers. However, peculiarities of language use make the text difficult to process, and the presence of sensitive information makes it hard to obtain adequate quantities for developing processing systems. The short term goal of most research in the area is to achieve a reliable language processing foundation that can support more complex tasks such as named entity recognition (NER) to a sufficiently reliable level.

Chunking is the task of identifying non-recursive phrases in text (Abney, 1991). It is a type of shallow parsing that is a less challenging task than dependency or constituency parsing. This makes it likely to give more reliable results on clinical text, since there is a very limited amount of annotated (or even raw) text of this kind available for system development. Even though chunking does not provide as much syntactic information as full parsing, it is an excellent method for identifying base noun phrases (NP), which is a key issue in symptom and disease identification. Identifying symptoms and diseases is at the heart of harnessing the potential of clinical data for medical research purposes.

There are few resources that enable researchers to adapt general domain techniques to clinical text. Using the Harvey Corpus[1] – a chunk annotated clinical text language resource – we present an exploratory study into adapting general domain tools and models to apply to free text notes typed by UK primary care physicians.

2 Related Work

The Mayo Clinic Corpus (Pakhomov et al., 2004) is a key resource that has been widely used as a gold standard in part-of-speech (POS) tagging of clinical text. Based on that corpus and the Penn TreeBank (Marcus et al., 1993), Coden et al. (2005) present an analysis of the effects of domain data on the performance of POS tagging models, demonstrating significant improvements with models trained entirely on domain data. Savova et al. (2010) use this corpus for the development of cTAKES, Mayo Clinic's processing pipeline for clinical text.

Fan et al. (2011) show that using more diverse clinical data can lead to more accurate POS tagging. They report that models trained on clinical text datasets from two different institutions perform on each of the datasets better than both models trained only on the same or the other dataset.

Fan et al. (2013) present guidelines for syntactic parsing of clinical text and a clinical Treebank annotated according to them. The guidelines are designed to help the annotators handle the non-canonical language that is typical of clinical text.

[1] An article describing the corpus is currently under review.

Proceedings of the 2014 Workshop on Biomedical Natural Language Processing (BioNLP 2014), pages 77–82,
Baltimore, Maryland USA, June 26-27 2014. ©2014 Association for Computational Linguistics

3 Data

The Harvey Corpus is a chunk-annotated corpus consisting of pairs of manually anonymised UK primary care physician (General Practitioner, or GP) notes and associated Read codes (Bentley et al., 1996). Each Read code has a short textual gloss. The purpose of the codes is to make it easy to extract structured data from clinical records. The reason we include the codes in the corpus is that GPs often use their glosses as the beginning of their note. Two typical examples (without chunk annotation for clarity) are shown below.

Birth details *|| Normal deliviery Girl Weight - 3. 960kg Apgar score @ 1min - 9 Apgar score @ 5min - 9 Vit K given Paed check NAD HC - 34. 9cm Hip test performed* (1)

Chest pain *|| musculoskel pain last w/e, nil to find, ecg by paramedic no change, reassured, rev sos* (2)

The corpus comprises 890 pairs of Read codes and notes, each annotated by medical experts using a chunk annotation scheme that includes non-recursive noun phrases (NPs), main verb groups (MVs), and a common annotation for adjectival and adverbial phrases (APs). Example (3) below illustrates the annotation. The majority of the records (750) were double blind annotated by medical experts, after which the resulting annotation was adjudicated by a third medical expert annotator.

*[**Chest pain**]*^{NP} *||* *[musculoskel pain]*^{NP} *[last w/e]*^{NP}*,* *[nil]*^{AP} *to* *[find]*^{MV}*,* *[ecg]*^{NP} *by* *[paramedic]*^{NP} *[no change]*^{NP}*,* *[reassured]*^{MV}*,* *[rev]*^{MV} *[sos]*^{AP} (3)

Inter-annotator agreement was 0.86 f-score, taking one annotator to be the gold standard and the other the candidate. We calculate the f-score according to the MUC-7 (Chinchor, 1998) specification, with the standard f-score formula. The calculation is kept symmetric with regard to the choice of gold standard annotator by limiting the counting of *incorrect* categories to one per tag, and equating the *missing* and *spurious* categories. For example, three words annotated as one three-token chunk by annotator A and three one-token chunks by annotator B will have one incorrect and two missing/spurious elements.

The rest of the records are a by-product of the training process. Ninety records were triple annotated by three different medical experts with the help of a computational linguist, and fifty records were double annotated by a medical expert – alone and together with a computational linguist.

It is important to note that the text in the corpus is not representative of all types of GP notes. It is focused on text that represents the dominant part of day-to-day notes, rather than standard edited text such as copies of letters to specialists and other medical practitioners.

Even though the corpus data is very rich in information, its non-canonical language means that it is very different from other clinical corpora such as the Mayo Clinic Corpus (Pakhomov et al., 2004) and poses different challenges for processing. The GP notes in the Harvey Corpus can be regarded as groups of medical 'tweets' meant to be used mainly by the author. Sentence segmentation in the classical sense of the term is often impossible, because there are no sentences. Instead there are short bursts of phrases concatenated together often without any indication of their boundaries. The average length of a note is roughly 30 tokens including the Read code. This is in contrast to notes in other clinical text datasets, which range from 100 to 400 tokens on average (Fan et al., 2011; Pakhomov et al., 2004). As well as typical clinical text characteristics such as domain-specific acronyms, slang, and abbreviations, punctuation and casing are often misleading (if present at all), and some common classes of words (e.g. auxiliary verbs) are almost completely absent.

4 Chunking

State-of-the-art text chunking accuracy reaches an f-score of 95% (Sun et al., 2008). However, this is for standard, edited text, and relies on accurate POS tagging in a pre-processing step. However, the characteristics of GP-written free text make accurate part of speech (POS) tagging and chunking difficult. Major problems are caused by unknown tokens and ambiguities due to omitted words or phrases.

We evaluate two standard chunking tools, YamCha (Kudo and Matsumoto, 2003) and CRF++[2], selected based on their support for trainable context features. The tools were applied to the Har-

[2] `http://crfpp.googlecode.com/svn/trunk/doc/index.html`

	POS	YamCha IOB	YamCha BEISO	CRF++ IOB	CRF++ BEISO
ARK$_{IRC}$	75.35	76.63 σ1.04	76.87 σ2.91	75.87 σ1.64	76.23 σ1.99
ARK$_{Twitter}$	–	**76.72 σ2.11**	**77.53 σ1.65**	**76.63 σ2.36**	**77.23 σ1.06**
ARK$_{Ritter}$	75.70	76.59 σ2.01	76.72 σ2.11	**76.63 σ1.05**	77.17 σ1.77
cTAKES	**82.42**	75.32 σ2.52	75.85 σ2.02	75.43 σ1.79	75.53 σ1.90
GENIA	80.63	71.70 σ2.27*	74.86 σ1.41	74.16 σ2.03*	74.19 σ1.72
RASP	–	74.24 σ1.84	75.10 σ1.31	75.63 σ2.33	75.76 σ2.18
Stanford	80.68	76.40 σ1.69	76.36 σ2.92	75.95 σ1.25	75.94 σ1.91
SVMTool	76.40	74.32 σ2.57	74.30 σ2.71	74.66 σ1.77	74.68 σ2.28
Wapiti	73.39	74.74 σ2.29	74.78 σ1.33	73.59 σ2.62	73.83 σ2.31
baseline	–	69.66 σ1.89*	69.76 σ1.24	67.05 σ1.15*	68.65 σ1.41

Table 1: Chunking results using *YamCha* and *CRF++* on data automatically POS tagged using nine different models; the baseline is with no tagging. The IOB and BEISO columns compare the impact of two chunk representation strategies. The POS column indicates the part-of-speech tagging accuracy for a subset of the corpus. Asterisks indicate pairs of significantly different YamCha and CRF++ results (t-test with 0.05 p-value).

vey Corpus with automatically generated POS annotation. Given the small amount of data and the challenges presented above, we expected that our results would be lower than those reported by Savova et al. (2010). The aim of these experiments is to find the best performance obtainable with standard chunking tools, which we will build on in further stages of our research.

We conducted pairs of experiments, one with each chunking tool, divided into three groups: the first investigates the effects of choice of POS tagger for training data annotation (Section 4.1); the second compares two chunk representations (Section 4.2); and the third searches for the optimal context features (Section 4.3). All feature tuning experiments were conducted on a development set and tested using 10-fold cross-validation on the rest of the data. We used 10% of the whole data for the development set and 90% of the remaining data for a training sample during development. This guarantees the development model is trained on the same amount of data as the testing model.

4.1 Part-of-Speech Tagging

We evaluated and compared the results yielded by the two chunkers, having applied each of seven off-the-shelf POS taggers. Of these taggers, cTAKES (Savova et al., 2010) and GENIA (Tsuruoka et al., 2005) are the only ones trained on data that resembles ours, which suggests that they should have the best chance of performing well. We also selected a number of other taggers while trying to diversify their algorithms and training data as much as possible: the POS tagger part of the Stanford NLP package (Toutanova et al., 2003) because it is one of the most successfully applied in the field; the RASP tagger (Briscoe et al., 2006) because of its British National Corpus (Clear, 1993) training data; the ARK tagger (Owoputi et al., 2013) because of the terseness of the tweet language; and the SVMTool (Giménez and Màrquez, 2004) and Wapiti (Lavergne et al., 2010) because they use SVM and CRF algorithms. Our baseline model uses no part of speech information.

Using the Penn TreeBank tagset (Marcus et al., 1993), we manually annotated a subset of the corpus of comparable size to the development set. Using this dataset we estimated the tagging accuracy for all models that support that tagset (omitting *RASP* and *ARK Twitter* since they use different tagsets). In this evaluation, cTAKES is the best performing model, followed closely by the Stanford POS tagger and GENIA.

The results in Table 1 show that the differences between chunking models trained on different POS annotations are small and mostly not statistically significant from each other. However, all the results are significantly better than the baseline, apart from those based on the GENIA tagger output.

4.2 Chunk Representation

The dominant chunk representation standard *inside, outside, begin* (IOB) introduced by Ramshaw and Marcus (1995) and established with the

CoNLL-2000 shared task (Sang and Buchholz, 2000) takes a minimalistic approach to the representation problem in order to keep the number of labels low. Note that for chunking representations the total number of labels is the product of the chunk types and the set of representation types plus the outside tag, meaning that for IOB with our set of three chunk types (NP, MV, AP) there are seven labels.

Alternative chunk representations, such as *begin, end, inside, single, outside* (BEISO)[3] as used by Kudo and Matsumoto (2001), offer more fine-grained tagsets, presumably at a performance cost. That cost is unnecessary unless there is something to be gained from a more fine-grained tagset at decoding time, because the two representations are deterministically inter-convertible. For instance, an *end* tag could be useful for better recognising boundaries between chunks of the same type. The BEISO tagset model looks for the boundary before and after crossing it, while an IOB model only looks after. This should give only a small gain with standard edited text because the chunk type distribution is fairly well balanced and punctuation divides ambiguous cases such as lists of compound nouns. However, the Harvey Corpus is NP-heavy and contains many sequences of NP chunks that do not have any punctuation to mark their boundaries.

We evaluated the two chunk representations in combination with each POS tagger. Table 1 shows that the differences between the results for the two representations are small and never statistically significant. We also evaluated the two chunk representations with different amounts of training data. The resulting learning curves (Figure 1) are almost identical.

4.3 Context Features

We approached the feature tuning task by first exploring the smaller feature space of YamCha and then using the trends there to constrain the features of CRF++. YamCha has three groups of features responsible for tokens, POS tags and dynamically generated (i.e. preceding) chunk tags. For all experiments we determined the best feature set by exhaustively testing all context feature combinations within a predefined range. We used the same context window for the token and tag features in order to reduce the search space. Given the terseness of the text we expected that wider context windows of more than three tokens would not be beneficial to the model, and therefore did not consider them. Our experiments using YamCha confirmed this hypothesis and showed a consistent trend among all experiments in favouring a window of -1 to +1 for tokens and slightly wider for chunk tags (see Table 2).

Feature Set	**CV**	**Dev**
W_{-1}-W_1, T_{-1}-T_1, C_{-1}	77.28 σ1.9	75.28
W_{-1}-W_1, T_{-1}-T_1, C_{-2}-C_{-1}	77.27 σ2.6	74.70
W_{-1}-W_2, T_{-1}-T_2, C_{-1}	76.86 σ1.5	74.08
W_{-2}-W_1, T_{-2}-T_1, C_{-2}	76.46 σ1.3	74.00
W_{-1}-W_1, T_{-1}-T_1, C_{-2}	76.89 σ2.1	73.92
W_{-2}-W_1, T_{-2}-T_1, C_{-3}-C_{-1}	76.52 σ0.9	73.91
W_{-1}-W_1, T_{-1}-T_1, C_{-3}-C_{-1}	77.02 σ2.0	73.90
W_{-2}-W_2, T_{-2}-T_2, C_{-1}	77.03 σ1.9	73.86
W_{-1}-W_1, T_{-1}-T_1, C_{-3}	77.15 σ1.5	73.63
W_{-3}-W_1, T_{-3}-T_1, C_{-2}-C_{-1}	75.71 σ1.9	73.63

Table 2: Development set and 10-fold cross-validation results for the top ten feature sets of YamCha models trained on ARK$_{\text{Twitter}}$ POS annotation. Token features are represented with *W*, POS features with *T*, and dynamically generated chunk features with *C*. None of the cross-validation results are significantly different from each other (t-test with 0.05 p-value).

CRF++ provides a more powerful feature configuration allowing for unary and pairwise[4] features of output tags. The unary features allow the construction of token or POS tag bigrams and trigrams in addition to the standard context windows. The feature tuning search space with so many parameters is enormous, which required us to use our findings from the YamCha experiments to trim it down and make it computationally feasible. First, we decreased the search window of all features by one in each direction from -3:3 to -2:2. Second, we used the top scoring POS model from the first experimental runs to constrain the features even further by selecting only the top one hundred for the rest of the models.

We could not identify the same uniform trend in the top feature sets as we could with YamCha. Our results ranged from very small context windows to the maximum size of our search space. How-

[3] Also sometimes abbreviated IOBSE

[4] The unary and pairwise features of output tags are referred to as *unigram and bigram features of output tags* on the CRF++ web page. Although this is correct, it can also be confused with unigrams and bigrams of tokens, which are expressed as unary (unigram) output tag features.

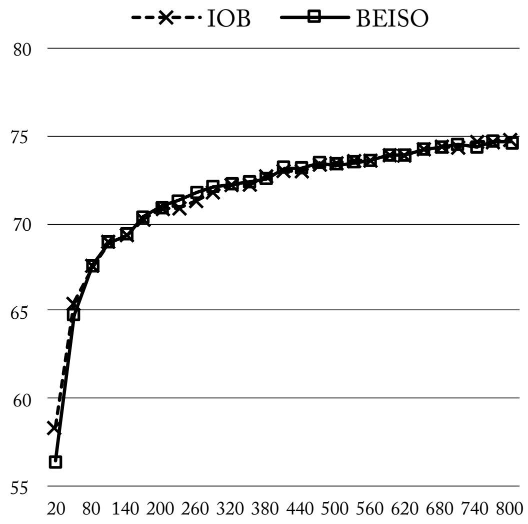

Figure 1: Chunking results for YamCha IOB and BEISO models with increasing amounts of training data.

ever, we noticed that BEISO feature sets tend to be smaller than the IOB ones. We also found that the pairwise features normally improve the results.

5 Discussion and Future Work

We were surprised that the experiments did not show a clear correlation between POS tagging accuracy and chunking accuracy. On the other hand, the chunking results using POS tagged data are significantly better than the baseline, except when using the GENIA tagger output. The small differences between training sets of similar POS accuracy could be explained due to the non-uniform impact of the wrong POS tag on the chunking process. Some mistakes such as labelling a noun as a verb in the middle of a NP chunk are almost sure to propagate and cause further chunking errors, whereas others may have minimal or no effect, for example labelling a singular noun as a proper noun. An error analysis of verb tags and noun tags (Table 3) shows that the ARK models tend to make more mistakes that keep the annotation within the same tag group compared to the GENIA model (see column pairs 1 and 3, and 2 and 4). This is a possible explanation for the lower accuracy of the chunking model trained on data tagged by GENIA.

Our experiments showed that the models using the two chunk representations did not perform significantly differently from each other. We also showed that this conclusion is likely to hold if more training data were available.

Model	N_{group}	V_{group}	Nouns	Verbs
ARK_{IRC}	67.17	78.26	88.26	85.99
$ARK_{Twitter}$	-	-	86.97	88.71
ARK_{Ritter}	68.57	77.29	90.64	85.02
cTAKES	83.93	62.80	93.85	69.08
GENIA	81.56	61.83	92.03	71.01
RASP	-	-	84.59	83.58
Stanford	80.30	73.42	91.89	83.09
SVMTool	69.97	70.04	90.08	80.19
Wapiti	65.64	66.66	87.84	74.87

Table 3: Detailed view of the POS model results focusing on the noun and verb tag groups. The leftmost two columns of figures show accuracies over tags in the respective groups; the rightmost two columns show the accuracies of the same groups if all tags in a group are replaced with a group tag, e.g. *V* for verbs[5].

There are a number of ways we could improve chunking accuracy besides increasing the amount of training data. Although our results do not show a clear trend, Fan et al. (2011) demonstrate that the domain of part-of-speech training data has a significant impact on tagging accuracy, which could potentially impact chunking results if it decreases the number of errors that propagate during chunking. An important problem in that area is dealing with present and past participles, which are almost sure to cause error propagation if mislabelled (as nouns or adjectives, respectively). Participles are more ambiguous in terse contexts lacking auxiliary verbs, which are natural disambiguation indicators. Another direction in processing that could contribute to better chunking is better token and sentence segmentation. Finally, unknown words, which may potentially have the largest impact on chunking accuracy, could be dealt with using a generic solution such as feature expansion based on distributional similarity.

References

S. Abney. 1991. Parsing by chunks. In Robert C. Berwick, Steven P. Abney, and Carol Tenny, editors, *Principle-Based Parsing: Computation and Psycholinguistics*, pages 257–278. Kluwer, Dordrecht.

T. Bentley, C. Price, and P. Brown. 1996. Structural and lexical features of successive versions of the

[5] Note that these results are different from what would be yielded by a classifier trained on data subjected to the same tag substitution.

read codes. In *Proceedings of the Annual Conference of The Primary Health Care Specialist Group of the British Computer Society*, pages 91–103.

T. Briscoe, J. Carroll, and R. Watson. 2006. The second release of the RASP system. In *Proceedings of the COLING/ACL on Interactive Presentation Sessions*, COLING-ACL'06, pages 77–80, Stroudsburg, PA, USA. Association for Computational Linguistics.

N. Chinchor. 1998. Appendix B: Test scores. In *Proceedings of the Seventh Message Understanding Conference (MUC-7)*, Fairfax, VA, April.

J. Clear. 1993. The British National Corpus. In George P. Landow and Paul Delany, editors, *The Digital Word*, pages 163–187. MIT Press, Cambridge, MA, USA.

A. Coden, S. Pakhomov, R. Ando, P. Duffy, and C. Chute. 2005. Domain-specific language models and lexicons for tagging. *Journal of Biomedical Informatics*, 38:422–430.

J.-W. Fan, R. Prasad, R.M. Yabut, R.M. Loomis, D.S. Zisook, J.E. Mattison, and Y. Huang. 2011. Part-of-speech tagging for clinical text: Wall or bridge between institutions? In *American Medical Informatics Association Annual Symposium*, 1, pages 382–391. American Medical Informatics Association.

J.-W. Fan, E. Yang, M. Jiang, R. Prasad, R. Loomis, D. Zisook, J. Denny, H. Xu, and Y. Huang. 2013. Research and applications: Syntactic parsing of clinical text: guideline and corpus development with handling ill-formed sentences. *JAMIA*, 20(6):1168–1177.

J. Giménez and L. Màrquez. 2004. SVMTool: A general POS tagger generator based on support vector machines. In *Proceedings of the 4th LREC*, Lisbon, Portugal.

T. Kudo and Y. Matsumoto. 2001. Chunking with support vector machines. In *Proceedings of the Second Meeting of the North American Chapter of the Association for Computational Linguistics on Language Technologies*, NAACL'01, pages 1–8, Stroudsburg, PA, USA. Association for Computational Linguistics.

T. Kudo and Y. Matsumoto. 2003. Fast methods for kernel-based text analysis. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 24–31, Morristown, NJ, USA. Association for Computational Linguistics.

T. Lavergne, O. Cappé, and F. Yvon. 2010. Practical very large scale CRFs. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 504–513. Association for Computational Linguistics, July.

M. Marcus, B. Santorini, and M. A. Marcinkiewicz. 1993. Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2):313–330.

O. Owoputi, B. O'Connor, C. Dyer, K. Gimpel, N. Schneider, and N. Smith. 2013. Improved part-of-speech tagging for online conversational text with word clusters. In *Proceedings of NAACL-HLT*, pages 380–390.

S. Pakhomov, A. Coden, and C. Chute. 2004. Creating a test corpus of clinical notes manually tagged for part-of-speech information. In *Proceedings of the International Joint Workshop on Natural Language Processing in Biomedicine and Its Applications*, JNLPBA'04, pages 62–65, Stroudsburg, PA, USA. Association for Computational Linguistics.

L. Ramshaw and M. Marcus. 1995. Text chunking using transformation-based learning. In *Proceedings of the Third Workshop on Very Large Corpora*, pages 82–94.

E. Sang and S. Buchholz. 2000. Introduction to the CoNLL-2000 shared task: Chunking. In *Proceedings of the 2nd Workshop on Learning Language in Logic and the 4th Conference on Computational Natural Language Learning*, pages 13–14.

G. Savova, J. Masanz, P. Ogren, J. Zheng, S. Sohn, K. Kipper-Schuler, and C. Chute. 2010. Mayo clinical text analysis and knowledge extraction system (cTAKES): architecture, component evaluation and applications. *Journal of the American Medical Informatics Association*, 17(5):507–513.

X. Sun, L.-P. Morency, D. Okanoharay, and J. Tsujii. 2008. Modeling latent-dynamic in shallow parsing: A latent conditional model with improved inference. In *Proceedings of the 22nd International Conference on Computational Linguistics (COLING'08)*, Manchester, UK, August.

K. Toutanova, D. Klein, C. D. Manning, and Y. Singer. 2003. Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1*, NAACL'03, pages 173–180, Stroudsburg, PA, USA. Association for Computational Linguistics.

Y. Tsuruoka, Y. Tateishi, J.-D. Kim, T. Ohta, J. McNaught, S. Ananiadou, and J. Tsujii. 2005. Developing a robust part-of-speech tagger for biomedical text. In *Proceedings of the 10th Panhellenic Conference on Advances in Informatics*, PCI'05, pages 382–392, Berlin, Heidelberg. Springer-Verlag.