IJCLA

ISSN 0976-0962

International Journal of Computational Linguistics and Applications

Vol. 4 No. 2 Jul-Dec 2013

Guest Editor
Efstathios Stamatatos

Editor-in-Chief
Alexander Gelbukh

ISSN 0976-0962

International Journal of Computational Linguistics and Applications

Vol. 4 No. 2 Jul-Dec 2013

International Journal of Computational Linguistics and Applications – IJCLA (started in 2010) is a peer-reviewed international journal published twice a year, in June and December. It publishes original research papers related to computational linguistics, natural language processing, human language technologies and their applications.

The views expressed herein are those of the authors. The journal reserves the right to edit the material.

Indexing: Cabell's Directory of Publishing Opportunities.

Editor-in-Chief: Alexander Gelbukh

Subscription:	India:	Rs. 2699
	Rest of the world:	US$ 249

Payments can be made by Cheques/Bank Drafts/International Money Orders drawn in the name of BAHRI PUBLICATIONS, NEW DELHI and sent to:

BAHRI PUBLICATIONS
1749A/5, 1st Floor, Gobindpuri Extension,
P. O. Box 4453, Kalkaji, New Delhi 110019
Telephones: 011-65810766, (0) 9811204673, (0) 9212794543
E-mail: bahrius@vsnl.com; bahripublications@yahoo.com
Website: http://www.bahripublications.com

Printed & Published by Deepinder Singh Bahri, for and on behalf of **BAHRI PUBLICATIONS**, New Delhi.

International Journal of Computational Linguistics and Applications

Vol. 4 No. 2 Jul-Dec 2013

CONTENTS

CO-REFERENCE RESOLUTION

MACHINE TRANSLATION

INFORMATION EXTRACTION AND BIOMEDICAL APPLICATIONS

NATURAL LANGUAGE GENERATION AND GRAMMAR CHECKING

Editorial

This issue of IJCLA presents papers on four topics: co-reference resolution; machine translation; information extraction and biomedical applications; and natural language generation and grammar checking.

The first section consists of one paper devoted to co-reference resolution, which is a process of automatically detecting whether two different words in the text refer to the same entity in real world. The simplest example are pronouns, but other words can also participate in co-reference: for example, *Barack Obama*, *the President*, and *he* can, in a suitable context, refer to the same person. Thus linking these words together is important for text understanding, as well as for many applications ranging from information retrieval and question answering to opinion mining and machine translation.

D. Weissenbacher and **Y. Sasaki** (France and Japan) study the approach to co-reference resolution with Bayesian networks. Different factors can affect the quality of the process of co-reference resolution in a machine learning framework. The most studied ones are feature selection and the learning algorithm used; others are less studied. The authors present a comprehensive study of various factors that affect this process, and conclude that two factors have important impact on its quality: how noisy the features used for classification are, and how reliably the algorithm detects whether a given word is a reference to some another word in the text. For example, in the text *it is clear that this idea is novel* the word *it* does not refer to any other word in the text, while in the text *the idea was difficult to understand but now it is clear* the word *it* refers to *the idea*; looking for an antecedent in the first case (and thus choosing the least unsuitable one) would result in an error.

The second section presents three papers devoted to machine translation. Automatic translation technologies are quickly coming of age and become part of our everyday life. They contribute to better understanding between people of different cultures in our globalized world and help people of all nations to integrate into global community.

X. Song *et al.* (UK) show how to better evaluate the results of machine translation algorithms. The standard automatic evaluation metric nowadays is BLEU, which, despite its usefulness, has certain limitations, such as its inability to handle very short texts—which are very common in Internet and social networks, as well as rather low agreement with human judgments. The authors propose a simpler variant of this evaluation metric that is more flexible and more reliable. They show that their proposed metric has better agreement with human judgments than the standard BLEW metric currently widely used for evaluation of machine translation systems.

G. Wisniewski and **F. Yvon** (UK) suggest a much faster training method for machine translation algorithms. Slow training is a bottleneck for development of statistical machine translation systems and for experimentation with the corresponding algorithms. The authors show that recent advances in recent advances in stochastic optimization and online machine learning can lead to significant improvement in training speed with competitive quality of the resulting translation.

L. Laki et al. (Hungary) present a rule-based method for reordering of phrases in phrase-based machine translation. Reordering is the most important issue that affects quality of phrase-based machine translation when the two languages have different structure and word order. On the example of English to Hungarian translation the authors show how the system can reorder the source sentences (English) in order to make them more similar to the expected translation in the target language (Hungarian) before actual translation. For example, an English phrase *the sons of the many merchants living in the city* is transformed to, roughly speaking, *the city-in living many merchants sons-of*, which is much closer to how the phrase is going to look in Hungarian, after which only a literal translation of English words is required to complete the process.

The next section consists of four paper devoted to information extraction, especially its biomedical applications. Information extraction is a process of automatically building databases and knowledge bases by extracting structured information—such as which medicine causes which side effect—from raw unstructured texts. This process requires significant degree of understanding both structure and semantics of the text.

S. Hina *et al.* (UK and Pakistan) present a semantic tagger for medical narratives, capable of tagging complex semantic information, including paraphrases, abbreviations, and multiword concepts. Such a tagger is useful for a wide variety of applications such as question answering or statistical analysis. The tagging process suggested by the authors is based on rule patterns identified from a real world medical dataset. The proposed tagger outperforms existing methods, including both SVM-based machine learning approach and ontology-based approach.

R. Nawaz *et al.* (UK) go beyond semantics to explore discourse structure of biomedical texts. Discourse-level analysis includes identification of discourse relations between text spans and rhetorical status of sentences and clauses. It is important for identification and interpretation of meta-knowledge: knowledge about knowledge. The authors show how to detect patterns of expressions that convey meta-knowledge about events in scientific papers. They also point out differences between such patterns in the full text of scientific papers and in their abstracts.

D. Kokkinakis (Sweden) continues the topic of extraction of medical events from text. He explores the possibility of using the Frame Semantics framework for this purpose, in particular, the large FrameNet lexical resource combined with domain-specific knowledge sources. He uses a rule-based approach, though machine-learning techniques can be later incorporated in the same framework. He shows that this approach provides powerful modeling mechanism for text mining and information extraction, with high quality of achieved results.

C. Li *et al.* (Hong Kong) propose a framework for named entity detection in Internet texts. Named entities are important in information extraction since they indicate the participants of relations to be extracted. The authors use an approach that does not require training labeled examples; instead, they leverage existing resources and dictionaries for training. Via extensive experiments they show the effectiveness of their approach.

Finally, the last section consists of three papers devoted to natural language generation and grammar checking, which are important applications of natural language techniques.

Y. Hayashi et al. (Japan) show how to determine correct sentence order in a text that consists of various sentences. The problem is important in style correction, where the system can suggest the user a better ordering of the sentences to make the text more understandable. It is also important in natural language generation, where the order of the sentences is to be decided before their actual generation. Natural language generation has a number of applications, of which multi-document summarization is currently the most important one. On the example of Japanese topic-marking particles the author show how linguistic information in a rule-based approach improves the results over the more widely used probabilistic approaches.

G. Sidorov (Mexico) continues the topic of importance of linguistic information for natural language processing tasks. He explains in detail the use of a newly introduced linguistic-based feature called syntactic n-grams in the task of grammar checking of English texts written by non-native speakers. Similarly to a number of other tasks, where the usefulness of the syntactic n-grams as machine-learning features have been already demonstrated, he shows that very simple system based on this approach can show performance competitive with much more sophisticated systems, thus once more confirming that syntactic n-grams are a very useful tool for diverse language processing tasks.

L. Cinman *et al.* (Russia) address the problem of assessing text quality not in the setting of style correction for human authors but instead in the setting of automatically distinguishing human-written texts from automatically generated ones. The problems is very important in fighting spam. What is more, while probably the majority of current natural language processing systems deal with Internet texts, webpages are often full of automatically generated contents usually useless for both applications and human readers, which leads to the necessity of so-called boilerplate removal: mining for useful content in the flood of such useless texts. Even more importantly, fake automatically generated reviews hinder the applications of opinion mining. The authors achieve 85% F-measure on distinguishing between automatically generated and human-written texts, which will be extremely useful in all mentioned applications.

This issue of IJCLA will be useful for researchers, students, software engineers, and general public interested in natural language processing and its applications.

GUEST EDITOR:

EFSTATHIOS STAMATATOS
ASSISTANT PROFESSOR,
UNIVERSITY OF THE AEGEAN, GREECE
WEB: < WWW.ICSD.AEGEAN.GR/LECTURERS/STAMATATOS>

Co-reference Resolution

IJCLA VOL. 4, NO. 2, JUL-DEC 2013, PP. 11–26
RECEIVED 05/12/12 ACCEPTED 04/03/13 FINAL 05/03/13

Which Factors Contribute to Resolving Coreference Chains with Bayesian Networks?

DAVY WEISSENBACHER AND[1] YUTAKA SASAKI[2]

[1] *IRISA, France*
[2] *Toyota Technological Institute, Japan*

ABSTRACT

This paper describes coreference chain resolution with Bayesian Networks. Several factors in the resolution of coreference chains may greatly affect the final performance. If the choice of machine learning algorithm and the features the learner relies on are largely addressed by the community, others factors implicated in the resolution, such as noisy features, anaphoricity resolution or the search windows, have been less studied, and their importance remains unclear. In this article, we describe a mention-pair resolver using Bayesian Networks, targeting coreference resolution in discharge summaries. We present a study of the contributions of comprehensive factors involved in the resolution using the 2011 i2b2/VA challenge data set. The results of our study indicate that, besides the use of noisy features for the resolution, anaphoricity resolution has the biggest effect on the coreference chain resolution performance.

KEYWORDS: *Coreference resolution, anaphoricity resolution, Bayesian networks, clinical informatics*

1 INTRODUCTION

Anaphora is a linguistic relation between two textual entities, which are commonly named mentions. The relation is defined when an entity, *the anaphor*, refers to another one, *the antecedent*. For example, in the following sentences:

[Mr. TTT]$_1$ *was brought to [the operating room]*$_2$ *where [he]*$_1$ *underwent [a coronary artery bypass graft]*$_3$ *x 3. [The patient]*$_1$ *tolerated [the procedure]*$_3$ *well.*

the pronoun *[he]*$_1$ is the anaphor, and it refers to the Noun Phrase (NP), *[Mr. TTT]*$_1$. When both mentions of the anaphoric relation refer to an identical object of the world, the relation is said to be coreference. As coreference is an equivalence relation, all mentions can be partitioned into different classes called *coreference chains*. In our example we have two coreference chains subscripted 1 and 3. The NP {*the operating room*} is a singleton and does not form a chain.

The resolution of coreference chains is still a difficult task. Whereas several factors are co-dependent in the resolution and may greatly affect the final performance when not set up correctly, only a few of them received specific attention in previous studies. While (1) the choice of the Machine Learning (ML) framework and (2) the features the ML algorithm relies on are largely addressed by the community, (3) the impact of the noise of the features, (4) the quality of the anaphoricity resolution and (5) the optimal size of the search windows, which are crucial in the mention-pair resolution strategy, have been less studied and their respective impacts on the resolution remain unclear.

The Informatics for Integrating Biology and Bedside (i2b2) institute has been holding a series of annual challenges to compare NLP systems on various tasks in the medical domain. The fifth i2b2/VA challenge, held in 2011, was on coreference resolution. While designing our own resolution system, we proceed to a comprehensive study of the effects of the above five factors on the overall performance of our system. The main contributions of this article are (1) to describe a mention-pair resolver based on a Bayesian Network addressing coreference resolution in discharge summaries and (2) to evaluate the direct effect of each factor on the overall resolution to guide further research by giving the highest priority to the most effective one.

The paper is organized as follows. In the following Section 2, we describe the resolver implemented and the features driving the classification. The corpus, the metrics and the protocol used for the experiments are detailed in Section 3. Impacts of the factors are discussed in Section 4. Section 5 presents related work, and finally Section 6 concludes the paper.

2 Resolving Coreference Chains

2.1 *Preprocessing*

To preprocess the i2b2/VA corpus, we use an annotation platform integrating publicly available annotation modules. It recognises the logical structures, *i.e.* titles, paragraphs, etc., thanks to handmade Regular Expressions (REs). As the sentence segmentation is crucial for anaphora resolution we used the pre-formatted sentences provided by the challenge organizers. To segment the words and produce a shallow parsing analysis of the documents (POS tagging and Chunking), we have chosen the Genia Tagger[3]. The pre-annotated concepts in the i2b2 corpora can be thought similar to Named Entities, we relied entirely on those concepts. The syntactic analysis of the sentences and the grammatical roles have been extracted by *Enju*[4]. Heads of NPs also play an important role in resolution since lots of features are computed based on them. To ensure good precision, NP and VP chunks are submitted and analysed separately from the whole sentence by *Enju*[5]. When the chunk analysis fails, heuristics are used [1]. Many resources have been developed for the Medical domain, we applied MetaMap[6] to automatically extract concepts of this domain.

2.2 *Resolution Strategy*

In a traditional approach to resolve coreference chains, two steps can be distinguished, the anaphoricity resolution followed by the coreference resolution.

Anaphoricity resolution consists of distinguishing anaphoric phrases which expect an antecedent from other phrases for which any suggestion of an antecedent would result in an error. Non-anaphoric phrases are, for example, pleonastic phrases (*e.g. It would be fine to...* vs *I have reviewed it...*), deictic phrases (*e.g.*, in our corpus, *this report, this year*) or the first NPs in coreference chains (first mentions of an object referred to by a chain are not anaphoric by definition).

The coreference resolution aims to build the coreference chains; all mentions referring to the same object should be included in a unique

[3] http://www-tsujii.is.s.u-tokyo.ac.jp/GENIA/tagger/

[4] http://www-tsujii.is.s.u-tokyo.ac.jp/enju/

[5] Extracting heads from the analysis of the full sentence gave bad results during preliminary experiments.

[6] http://metamap.nlm.nih.gov/

chain. When a strategy based on clustering is not chosen, a strategy relying on a binary classification is possible. For each anaphoric mention, considered in order, a list of previous mentions occurring in a search window is created, and one candidate in the list is chosen as antecedent. In the usual model, the *mention-pair* model, only pairs composed of an anaphoric mention and its respective candidates are described. Each pair received a score, and the candidate of the best pair is taken as antecedent. Once all pairs have been resolved, chains are built during an additional process, usually by taking the transitive closure with respect to the semantic constraints within the chains.

Classification methods are easy to use with the mention-pair model. We chose this model for our system. To build the coreference chains we took the transitive closure of the coreferent pairs. Incoherences within the resulting chains are post-edited by taking in the list of scored pairs the candidate of the first pair which agrees with the semantic constraints of the chain.

2.3 *Features and Classifiers*

In our system, pairs of mentions are described with a set of 32 features. They are features commonly used for coreference resolution plus features specific to the genre of our documents.

Our features can be separated into 3 categories: lexical, syntactic and semantic. Lexical features aggregate information about number, gender, position and all matching based features (string matching, embedded NPs, repeated NP etc.). Syntactic features provide information about grammatical roles of the mentions, syntactic parallelism or collocation patterns. Ground truth mentions annotated in the corpus are classified into 5 types of concepts: *person, problem, treatment, test, other*. From these semantic annotations we acquire reliable features and express constraints of coherence. Among the mentions denoting persons we specify, using handmade REs, the main protagonists of the discharge, namely the patient, his/her family, doctor and medical services. Mentions which do not refer to people are described in greater detail based on the MetaMap categories they match.

Pronouns have separate resolution procedures as they carry different information than NP mentions and tend to resolve with the closer candidates. We make use of 23 of the previous features to model the salience of the candidates as described in [1], except for the pronouns “I” and “we” which, in our corpus, are likely to resolve with the closest mention of the doctor.

To carry out the classification we select the Bayesian Network (BN) framework, a Machine Learning framework adapted to the distinctive characteristics of Natural Language Processing (NLP) tasks [2]. A BN is a probabilistic graphical model. It is composed of a qualitative description of the dependencies between a set of random variables, represented by an oriented acyclic graph, and of a quantitative description, a set of Conditional Probability Tables (CPT) where each random variable is associated to a graph node. For each of the previous features a random variable is created and the conditional probability table associated to the random variables gives information about which features it influences and is influenced by. In all our experiments the structure of the graph and the values of the CPTs are automatically learned from the data.

Because a coreference relation is an equivalence relation, positive and negative examples submitted to the machine learner during induction have to be carefully selected [3, 4]. Positive examples are anaphoric mentions linked to their closest immediate mention which belong in the same coreferent chain. A negative example is a pair of 2 mentions belonging to 2 different chains. We removed the trivial negative examples and presented only the 3 best negative examples for each anaphor to the system. The best negative examples are obtained during a preprocessing stage. The BN is trained iteratively 3 times, using the best pairs of the previous iteration.

As a first working hypothesis, our BN has been trained using the score-based algorithm K2 with a local metric, limited to 5 parents without imposing the Naive Bayes structure, combined with a *maximum a posteriori estimation*, the alpha parameter set to 0.5.

3 EXPERIMENTS

3.1 *Corpus and Metrics*

The corpus released for the 2011 i2b2/VA challenge is composed of discharge summaries provided by four health institutes. We worked with a subpart of the corpus, 251 documents for training and 173 for testing, referred to by the organizers as the i2b2/UPMC in [5].

To evaluate our system and compare it with other participants' systems, we used the official evaluation tool. By comparing the chains proposed by our system with the gold standard, this tool calculates 3 different metrics and their unweighted average: B^3, MUC, and CEAF. A presentation of those metrics and a discussion about their respective deficiencies can be found in [5].

3.2 *Protocol*

When the mention-pair strategy is applied, five factors can directly influence the performance of the coreference resolution. The first factor is the choice of the features to describe mentions and pairs. When the features are computed automatically, the second factor is the noise of the feature values. This type of noise can strongly degrade the induction process. As shown in [1] it might be better to do without a feature if it cannot be computed above a certain threshold of accuracy. Once the most reliable features have been selected, the machine learning framework, the third factor, has to be accurately chosen to ensure a good compromise between the power of expression required to learn the rules and the search for the optimal solution in the corresponding hypothesis space. The fourth factor is the choice of a strategy for resolving the anaphoricity. Whereas the anaphoricity resolution and the coreference resolution are co-dependent, the former has only lately received interest from the community [3]. The last factor is the size of the search window. It determines which mentions should be inserted in the list of possible candidates; the optimal size can never be known in advance since it depends on the genre and the domain of the corpus.

To optimise our coreference resolver we run a set of experiments, changing the value of one factor at a time, in order to find the more effective factors for the resolution. The next section presents the results obtained for each factor.

4 Evaluation

Noisy Features Factor. The impact of noisy features in resolution has been studied during the i2b2/VA challenge with a special track (track 1A). This track evaluates end-to-end resolution systems. With ground truth mentions hidden from the resolvers, a drop in performance of the systems ranging from 10.3% to 39.0% has been observed [5]. Noisy features appear to be the most critical factor to perfect in order to achieve a suitable score in coreference resolution. We did not carry out additional experiments for this factor.

Machine Learning Framework Factor. With the current progress of Machine Learning, many frameworks are now available, making the choice of a particular framework difficult. Their advantages depend on the number, the type and the structure of the data points the induction has to be run on. In the experiment below we intended to estimate the scale of the gain (or loss) by changing from one framework to another (even if some limitations have to be supported to use a particular framework). We have selected 4 frameworks broadly used in NLP: a decision tree classifier, a SVM classifier, a Naive Bayes and Bayesian Network classifiers.

For this experiment, all systems have their factors parametrised identically except for the classifiers they rely on to score the pairs of mentions[7]. All anaphoric mentions are given to the coreference resolver. The windows size is the largest possible, all anaphoric mentions which occur before the anaphoric mention to resolve are considered. To estimate the improvement, we report the performance of the baseline resolver published in [5]. The baseline resolver predicts all mentions as singletons.

Table 1 is quite revealing in two ways. First, it shows that there is a benefit of using an adapted ML framework. If all ML frameworks outperform the baseline system, there is a big difference in performance between the SVM and the BN, 7.8% in F-measure. The features used to model NLP data are strongly dependent due to the nature of Natural Language itself. The BN is the only classifier able to represent those dependencies and consequently makes a better discrimination between the mentions. The Naive Bayes classifier, helped by its knowledge of the prior probability of the features, is less sensitive to the missing values which are frequent with the features used for coreference resolution (*e.g.* unknown gender or grammatical roles). The default polynomial kernel support vector machine classifier proposed in Weka for the SVM classifier gives deceiving results, unusually worse than the decision tree. Better parameters or dedicated kernels should allow better results.

Secondly, the score of the BN, F=0.921, is higher than the score of the best system of the i2b2/VA challenge F=0.913 (P=0.905, R=0.920)[6], whereas our system does not make as extensive use of domain dependent knowledge as the latter system does. This fact supports the conclusion in [7] and is important since it demonstrates that an acceptable score can be achieved on this corpus using domain independent knowledge. How-

[7] We use the Weka machine learning tools. Each machine learning framework can be tuned to improve the induction, but we used the default options, except for the Bayesian Network where the default option is the Naive Bayes structure.

Table 1. Coreference resolution on Test corpus with various machine learning frameworks (anaphoric mentions are revealed, search window set to all previous candidates)

Systems	P	R	F
Baseline	.523	.602	.548
Decision Tree	.859	.850	.854
SVM	.849	.839	.843
Naive-Bayes	.894	.912	.903
Bayesian Network	**.912**	**.930**	**.921**

ever this result is only possible if the anaphoric mentions are perfectly resolved by the system. This perfect resolution is for the moment out of reach, even though resolving anaphoricity is much easier than resolving coreference [8], see section 4 for further discussion.

Feature Selection Factor. Features are central for the resolution because they express constraints/preferences to choose/discard a mention as antecedent. Therefore, they are the main subject of study for the coreference resolution. According to Zheng and *al.* [3] their number may largely vary within the range from 8 to 134. Their nature also is still under discussion: domain dependent features *vs.* general features.

In this study we give preference to domain independent features supplemented by semantic features adapted to the specific genre of our documents. The discharge summaries follow a specific scenario. A main actor, the patient, interacts with a few other characters, Doctor or medical services for instance, and whose body is described in detail. This causes a predominant chain of coreference, the chain of the patient, and several short coreferent chains. As for other participants' systems, our system relies on the categories associated with the mentions and tries to refine those categories. For the person category we wrote REs to discriminate the patient from the doctor, the family and medical services (*Coherent Roles* features in Table 2). To refine other categories we use the best UMLS concepts assigned by the word sense disambiguation module of the MetaMap tool (*Coherent Medical Concepts* features). Finally, we use the likelihood computed on the training corpus for two heads of mentions to be coreferent (*Heads Coreferent Mentions* features). Like Rink and *al.* [7], we believe this strategy can be applied to all documents with similar scenarios (accident reports or encyclopedia articles, are possible examples).

Table 2. Ablation study on the features used by the BN, performed on the Test corpus (anaphoric mentions are revealed, search window set to all previous candidates)

Bayesian Network	P	R	F
Lexical Feature	.927	.927	.927
Syntactic Feature			
+ Grammatical Roles	.910	.907	.909
+ Syntactic Parallelism	.910	.907	.909
+ Simple Collocations	.905	.903	.904
+ Syntactic Collocations	.902	.903	.902
Semantic Feature			
+ Coherent NEs	.902	.899	.900
+ Coherent Roles	.907	.905	.906
+ Coherent Medical Concepts	.912	.929	.921
+ Heads Coreferent Mentions	.912	.930	.921

The ablation study in Table 2 confirms the contribution of each feature. It suggests that the set of features added does not induce an important improvement of the overall score, only 2.8%. A similar conclusion can be drawn from Xu and *al.*'s [6] experiment. The best score is achieved by the lexical feature based system. Adding syntactic features does not improve the resolution and may even degrade the performance[8]. Semantic features improve the recall, particularly of medical concepts, but it is at the cost of a lower precision.

Anaphoricity Accuracy Factor. The good performance of our system is mainly due to the perfect anaphoricity resolution. To calculate its impact on the coreference resolution we introduced noise in the anaphoricity resolution. The anaphoricity resolver decides if a particular mention admits an antecedent or not; it does not have to find which mention is the antecedent. The quality of this resolution is crucial. False positives are mentions resolved as anaphoric when they are not and cause the coreference resolver to create new chains or include the false positives in any existing chain. False negatives are anaphoric mentions not recognized as such by the resolver and may result in a drop of recall if these anaphoric mentions are not chosen as antecedents for other anaphoric mentions.

[8] However syntactic features seem to corroborate the semantic ones. When our BN exploits lexical and semantic features without the syntactic ones, it performs worse than when it exploits all features, F=0.901 against F=0.921, respectively.

The current state-of-the-art scores range around 80% accuracy for a general domain corpus [4]. In order to evaluate the easiness of the task on our corpus, we have implemented a baseline anaphoricity resolver. Due to space limitations, we will not describe the anaphoricity resolver in detail. This resolver is also based on a Bayesian Network and performs two different resolutions for Definite NPs and for pleonastic pronouns *it, this, that, what, which* (other pronouns in our resolution are always considered anaphoric).

To classify a given Definite NP, features used are targeting possible synonyms which occur before the NP in the document. The synonyms are found based on string matching, edit distance, the WordNet dictionary, the MetaMap concepts of both mentions and the sections where possible synonyms appear. Sections are important in the discharge summaries since they indicate how to interpret following paragraphs, a context which is mandatory to resolve some coreferences. This can be illustrated briefly by two occurrences of *CVA* appearing in the section *History of Present Illness* and the section *Family History*; they are synonyms but they cannot be coreferent. Pleonastic pronouns *it* and *this* are detected by the filter described in [1] and adapted for our corpus. Other pleonastic pronouns are classified according to their immediate context. A pronoun, like the pronoun *what*, when immediately preceded by a noun tends to be anaphoric, whereas preceded by a verb is more likely to be non-anaphoric.

Despite its simplicity, our anaphoricity resolver reaches a decent score of 87.6% accuracy on the test corpus. Preliminary investigation of the results shows that the number of false negatives is much higher than the number of false positives, 2516 against 881. This is mainly due to the lack of the resources which are needed to establish synonym links between acronyms (such as *transesophageal echocardiogram* and *TEE*), hyperonyms (*examination* and *endoscopy*) or drug names (*lipitor* and *Atorvastatin*). General lexical resources such as Wikipedia have been found to be valuable resources [9, 7, 6] to provide such knowledge to the resolver.

Table 3 presents the coreference resolution achieved with varying quality of anaphoricity resolutions. According to predefined thresholds, we have corrupted gold anaphoric mentions to non-anaphoric and vice-versa. Mentions have been chosen randomly except for those which are preceded by a mention which exactly matches or has a similar head. Last constraints hold to avoid to corrupt anaphoric mentions which can be detected with a high precision. Bold scores are the score obtained when using the outputs of our anaphoricity resolver.

Table 3. Coreference resolution performances on the Test corpus for the BN given various anaphoricity resolutions *(in Accuracy)*

BN Performances	P	R	F
noise level			
0%	.912	.930	.921
5%	.913	.877	.895
10%	.892	.828	.857
13.4%	**.829**	**.891**	**.857**
15%	.881	.784	.826
20%	.862	.746	.794

From the data in Table 3 it is apparent that the biggest improvement is made by ameliorating the anaphoricity resolution with a possible gain of 12.7% in F-measure. Given the current performance of our anaphoricity resolver, 13.4% error rate, our coreference resolver reaches the top performance achieved during the last i2b2/VA challenge, with a score which is about equal to the score of the 9^{th} system of the competition (a total of 20 teams participated in).

Surprisingly, our system obtains a similar score when the noise threshold of is set to 10%. A possible explanation for this might be that in our experiment errors are randomly distributed, regardless of the easiness of the anaphoricity resolution. Whereas mentions incorrectly classified by our anaphoricity resolver are often the most difficult mentions to assign in chains.

Search Window Factor. The last factor is the size of the search window. The bigger the size of the window is, the higher is the risk to choose a “better” candidate, that is, a candidate different from the antecedent. While if the window is too small, none of the coreferent mentions may be found in the list of the candidates. The optimal size depends on the genre and the domain of the corpus [10]. In the discharge summaries, a list of medications or medical history report may separate an anaphor from its coreferent mentions by hundreds of sentences. The highest distance found in the training corpus was 274 sentences.

We have computed the search window as a percentage of sentences which have to be explored before finding the closest coreferent mention of each anaphoric mention. The ratios of antecedents captured by the

Table 4. Coreference resolution performance on the Test corpus for the BN given various search window sizes

BN performances with different search windows		P	R	F
Window size	*Antecedents captured*			
41%	94.55%	.906	.908	.907
67%	99.04%	.925	.926	.925
73%	99.62%	.926	.926	.926
90%	100%	.928	.929	.929
10 sentences with antecedents appended		.918	.934	.931

search windows have been computed directly on the test corpus[9]. Supplementary analysis shows that 20.3% are intrasentential anaphora in the test corpus (*resp.* 22.8% in the training corpus), 50.4% of the antecedent are located in the previous sentence (*resp.* 54.3%) and, as suggested by Zheng and *al.* [3], if the window is fixed as usual to the 10 previous sentences only 76.3% (*resp.* 79.2%) of the antecedents could have been found.

Table 4 summarizes the performance of the coreference resolver according to various sizes of windows. It appears that optimizing the size of the search window improved the performance of the resolution. Whereas the recall of the system sees no change, the precision, in reducing the number of candidates, has a consequent rise of 1.6%. This leads to the overall improvement of the system which does slightly better than the lexical based resolver described in Section 4.

However examining such proportion of document is still not satisfying. Many algorithms, for example based on centering[11] or the attention of the reader [12], have been proposed to update dynamically the list of candidates by removing from it impossible or old candidates. To test the interest of such algorithms we run a last experiment. We fixed a smaller size for the search window, set to the 10 previous sentences, and we artificially introduced the last coreferent mention. This experiment evaluates the capacity of the resolver to choose the coreferent mention among a few candidates and it suggests maximum scores reachable for the coreference resolution with our current features. With this last configuration the system's score reaches F=0.931.

[9] Similar computations on the training corpus have been done and show a difference of 7%. That is, a window of 83% on the training corpus is enough to capture all antecedents.

5 RELATED WORK

Our system is inspired from earlier modular strategies for resolution proposed by Rich and LuperFoy [13] or Mitkov [10]. Our approach targeting the patient and specialising other mention types is close to the general approach taken by the competing systems during the i2b2/VA Challenge [5]. Many of our features are similar to those described in [14].

Effects on the coreference resolution of several factors discussed in this article have been the main focus of several existing studies. While [15] examines possible discriminant features for clinical documents, the choice of features is still a significant problem for coreference resolution [16], [10] tests the benefits of using heuristics when the features are not available. Induction performed through various ML frameworks is studied by [17] for supervised methods. Advantages of sophisticated models compared to pairwise model resolution are criticized by Bengtson and Roth [4]. Finally, during their study to predict the difficulty of the coreference resolution on corpora, Stoyanov and *al.* [18] investigate possible performance improvements allowed by a better anaphoricity resolution and a better detection of the mentions. However, those studies often made comparisons between systems which differ by several factors at a time. In his extensive study about anaphora resolution, [10] draws our attention to the difficulties for making direct comparison between two coreference resolvers. If the systems are usually working on the same corpus, the preprocessing and the implementation of the features, for example, are rarely similar and introduce bias in the comparison. We are not aware of any existing study which carries out an exhaustive enquiry on the role of each factor for a given resolver. This article is an attempt to clearly measure the influences of the most important factors in the resolution.

6 CONCLUSION

In this article we introduced a promising coreference resolver based on a Bayesian Network and we presented a comprehensive study of the contribution of all important factors involved in the resolution.

Our system, to resolve coreference relations in clinical documents, relies on the mention-pair resolution strategy and uses a Bayesian Network to score the anaphoric pairs. The set of features implemented are features commonly used by ML based systems, completed with semantic features specialized for the genre of our documents. The semantic features track down the main objects of the discourse and express constraints by specifying the concepts these objects belong to. Using a basic anaphoricity resolver, we achieved an F-score of 0.857 on the 2011 i2b2/VA Challenge data set on coreference resolution.

By investigating the factors that contribute to the coreference resolution, our intention was to give a precise evaluation of their individual contributions to overall performance. Besides the use of noisy features for resolution, anaphoricity resolution has the biggest effect on the performance since both resolutions are strongly co-dependent. The choice of the ML framework can also strongly affect the results. The genre of the documents necessitate to adapt the size of the search window. Finally, the choice of the features, while main interest of the community, appears to be the less important factor in term of possible gain for resolution.

These findings suggest several courses of action for further enhancement of our resolver, with first priority given to our anaphoricity resolver. Based on Wikipedia, we are currently studying analogy distances between two mentions. By capturing valuable synonym relations this addition not only may largely improve our anaphoricity resolver, but also the coreference resolver. In the short-term the BN used for the anaphoricity resolution will be merged with the BN used for the coreference resolution in order to determine jointly both resolutions [8]. At medium term we will make use of Bayesian Logic Programs capable of representing all mentions and their associated chains within a unique probabilistic model, abolishing thus the unjustified independence assumption between the candidates, an assumption imposed by the current BN framework.

ACKNOWLEDGMENTS

We thank Pr. Gina-Anne Levow for her helpful comments and remarks.

REFERENCES

1. Weissenbacher, D., Nazarenko, A.: Comprendre les effets des erreurs d'annotations des plates-formes de tal. Traitement Automatique des Langues **52**(1) (2011) 161–185
2. Behera, L., Goyal, P., McGinnity, T.: Application of Bayesian Framework in Natural Language Understanding. IETE Technical Review **25**(5) (2008) 251–269
3. Zheng, J., Chapman, W., Crowley, R., Savova, G.: Coreference resolution: A review of general methodologies and applications in the clinical domain. Journal of Biomedical Informatics **44**(6) (2011) 1113–1122

4. Bengtson, E., Roth, D.: Understanding the value of features for coreference resolution. In: EMNLP'08. (2008)
5. Uzuner, O., Bodnari, A., Shen, S., Forbush, T., Pestian, J., South, B.: Evaluating the state of the art in coreference resolution for electronic medical records. Journal of the American Medical Informatics Association (2011)
6. Xu, Y., Liu, J., Wu, J., Wang, Y., Tu, Z., Sun, J., Tsujii, J., Chang, E.I.C.: A classification approach to coreference in discharge summaries: 2011 i2b2 challenge. Journal of the American Medical Informatics Association (2012)
7. Rink, B., Roberts, K., Harabagiu, S.: A supervised framework for resolving corerference in clinical records. Journal of the American Medical Informatics Association (2012)
8. Denis, P., Baldridge, J.: Joint determination of anaphoricity and coreference resolution using integer programming. In: Proceedings of NAACL. (2007) 236–243
9. Gooch, P., Roudsari, A.: Lexical patterns, features and knowledge resources for coreference resolution in clinical notes. Journal of Biomedical Informatics (2012)
10. Mitkov, R.: Anaphora Resolution. Longman(Pearson Education) (2002)
11. Grosz, B., Weinstein, S., Joshi, A.: Centering: a framework for modeling the local coherence of discourse. Computational Linguistics **21**(2) (1995) 203–225
12. Strube, M.: Never look back: An alternative to centering. In: 17th International Conference on Computational Linguistics. Volume 2. (1998) 1251–1257
13. Rich, E., LuperFoy, S.: An architecture for anaphora resolution. In: Proceedings of the second conference on Applied natural language processing. (1988) 18–24
14. Zweigenbaum, P., Wisniewski, G., Dinarelli, M., Grouin, C., rosset, S.: Résolution des coréférences dans des comptes rendus cliniques. Une expérimentation issue du défi i2b2/VA 2011. In: Actes de RFIA. (2012)
15. He, T.: Coreference resolution on entities and events for hospital discharge summaries. Master's thesis, MIT (2007)
16. Preiss, J.: Machine learning for anaphora resolution. Technical Report CS-01-10, University of Sheffield, Sheffield, England (2001)
17. Ng, V.: Supervised noun phrase coreference research: The first fifteen years. In: 48th Annual Meeting of the ACL. (2010) 1396–1411
18. Stoyanov, V., Gilbert, N., Cardie, C., Riloff, E.: Conundrums in noun phrase coreference resolution: Making sense of the state-of-the-art. In: Proceedings of the 47th Annual Meeting of the ACL. (2009) 656–664

DAVY WEISSENBACHER
IRISA (INRIA, UNIVERSITY OF RENNES 2, INSA, CNRS),
RENNES, FRANCE
E-MAIL: <DAVY.WEISSENBACHER@GMAIL.COM>

YUTAKA SASAKI
COIN LABORATORY,
TOYOTA TECHNOLOGICAL INSTITUTE,
2-12-1 HISAKATA, TEMPAKU, NAGOYA, 468-8511, JAPAN
E-MAIL: <YUTAKA.SASAKI@TOYOTA-TI.AC.JP>

Machine Translation

IJCLA VOL. 4, NO. 2, JUL-DEC 2013, PP. 29–44
RECEIVED 08/12/12 ACCEPTED 04/03/13 FINAL 05/03/13

BLEU Deconstructed: Designing a Better MT Evaluation Metric

XINGYI SONG, TREVOR COHN, AND LUCIA SPECIA

University of Sheffield, UK

ABSTRACT

BLEU is the de facto standard automatic evaluation metric in machine translation. While BLEU is undeniably useful, it has a number of limitations. Although it works well for large documents and multiple references, it is unreliable at the sentence or sub-sentence levels, and with a single reference. In this paper, we propose new variants of BLEU which address these limitations, resulting in a more flexible metric which is not only more reliable, but also allows for more accurate discriminative training. Our best metric has better correlation with human judgements than standard BLEU, despite using a simpler formulation. Moreover, these improvements carry over to a system tuned for our new metric.

1 INTRODUCTION

Automatic machine translation evaluation metrics provide a cheaper and faster way to evaluate translation quality than using human judgements. The standard evaluation metric in machine translation (MT) is BLEU [1], which is a simple, language independent metric that has been shown to correlate reasonably well with human judges. It is not only used in evaluation, but is also commonly used as a loss function for discriminative training [2, 3].

BLEU was designed for evaluating MT output against multiple references, and over large documents. However, evaluating translations at sentence level with single a reference is very common in MT research. Popular evaluation campaigns such as those organised by the WMT workshop only provide one reference for test and development corpora. In addition, many state-of-the-art discriminative training algorithms require sentence level evaluation metrics [4–6]. Often this means using a sentence-based approximation of BLEU, which can unduly bias the system and affect overall performance. BLEU performs less well when applied at the sentence level or sub-sentence level, and when using only one reference [7–10]. One reason is that in this setting BLEU has many zero or low counts for higher (tri-gram or higher) n-grams, and this has a disproportional effect on the overall score. Other problems with BLEU include its brevity penalty which has been shown to be a poor substitute for recall [10, 7], and the clipping of n-gram counts such that they do not exceed the count of each n-gram in the references, which complicates sub-sentential application.

Previous research has sought to address these problems. [11] suggest using arithmetic average instead of geometric mean. [12] shows that uni-gram and bi-gram precision contribute over 95 percent of overall precision, and they also state that adding higher order n-gram precision introduces a bias towards fluency over precision. This led us to question the effect of removing or substituting some components especially for sentence level evaluation. In this paper, we provide experimental analysis of each component in BLEU aiming to design better evaluation metrics for sentence level MT evaluation and MT system tuning with a single reference. On the WMT 2012 evaluation workshop [13], our variant of BLEU had better correlation with human judgements than any other for out-of-English document level evaluation.

The remainder of this paper is structured as follows: We will give brief a review of BLEU and its limitations in Section 2. In Section 3 we present experiments testing different variants of BLEU against human evaluation data, and also optimise the MT system parameters using these variant metrics. We found that our simplified BLEU improves over standard BLEU in terms of human judgements in both cases.

2 BLEU Review

The rationale behind BLEU [1] is that high quality translations will share many n-grams with human translations. BLEU is defined as

$$BLEU = BP \times \left(\prod_{n=1}^{4} p_n\right)^{\frac{1}{4}} \tag{1}$$

where p_n measures the modified n-gram precision between a document with candidate translations and a set of human authored reference documents, and the brevity penalty (BP) down-scales the score for outputs shorter than the reference. These are defined as

$$p_n = \frac{\sum\limits_{C \in \{Candidates\}} \sum\limits_{n-gram \in C} Count_{clip}(\text{n-gram})}{\sum\limits_{C' \in \{Candidates\}} \sum\limits_{n-gram' \in C'} Count(\text{n-gram'})} \tag{2}$$

$$BP = \begin{cases} 1 & \text{if } c > r \\ \exp(1 - \frac{r}{c}) & \text{if } c \leq r \end{cases}$$

where $Candidates$ are the set of sentences to be evaluated, c are their aggregate length and r is the length of the reference. $Count(\text{n-gram})$ counts the number of times the n-gram appears in the candidate sentence, and $Count_{clip}(\text{n-gram})$ is the same albeit clipped such that it does not exceed the number of times it appears in one of the reference sentences (which may be zero).

We now look at each of BLEU's component in detail.

N-gram precision BLEU is a precision-oriented evaluation method. Each precision component measures the proportion of predicted n-grams of a given n that appear in the reference translation. If multiple-references are used, the count of n-gram matching is based on the maximum number of matches against any of the references. For example in Table 1, candidate 1 matches 'It is a guide to action' and 'ensure that the military' with reference 1, matches 'which', 'always' and 'the commands of the party.' with reference 2. Therefore, the uni-gram precision will be 18/19, as only the word 'obeys' is not found in any of the references.

Clipping Clipping aims at penalising over-generated reference words in the candidate translation, such that repetitions of a word will not be rewarded. For example, candidate 2 in Table 2 is not a good translation, but still has very high uni-gram score (8/8). Clipping limits the count of a candidate n-gram to the maximum count of the n-gram in references. In this case the clipped uni-gram precision for candidate 2 will be 4/8: only one 'there' and one 'is' are treated as correct, and the repeats are counted as errors.

Brevity Penalty BLEU does not consider recall explicitly. In order to ensue reasonable coverage of reference, an alternative to recall is used: the

Table 1. Example of candidate and reference translations, adapted from [1].

Candidate 1:	It is a guide to action which ensures that the military always obeys the commands of the party.
Reference 1:	It is a guide to action that ensures that the military will forever heed Party commands.
Reference 2:	It is the guiding principle which guarantees the military forces always being under the command of the Party.
Reference 3:	It is the practical guide for the army always to heed the directions of the party.

Table 2. Without clipping and brevity penalty, candidates 1–3 will have same uni-gram score. Example taken from [1].

Reference:	there is a cat on the blue mat
Candidate 1:	there is
Candidate 2:	there there there is is is a cat
Candidate 3:	the cat is on the blue mat

brevity penalty. For example, candidate 1 in Table 2 has a uni-gram precision of 1. [1] state that in the multiple reference case, different words may be used in each reference, which makes it difficult to measure recall (we can never expect a good translation to include all these words). Therefore the Brevity Penalty is used instead to penalise short sentences. The intuition is that the candidate should have a similar length to the reference(s), and shorter candidates will be missing information.

2.1 *BLEU Limitations*

BLEU has become the standard evaluation metric since it was introduced in 2002, but it has several limitations. Firstly, in a short document or sentence, there is a high probability of obtaining zero tri-gram or 4-gram precision, which makes the overall BLEU score equal zero due to the use of geometric mean. Similarly, very low but non-zero counts disproportionately affect the score. A common method to ameliorate this effect is smoothing the counts [14–17], e.g. adding α both to the numerator and denominator of Equation 2. This avoids zero precision scores and zero overall BLEU score. However, different α values will affect the accuracy of the approximation of BLEU, and it is unclear what is a reasonable value to use. [11] suggest that using arithmetic average rather than geometric average, which avoids the problems of zero BLEU scores without resort to smoothing.

BLEU supports multiple references, which makes it hard to obtain an estimate of recall. Therefore, recall is replaced by the BP, but [10] state that BP is a poor substitute for recall. [10, 18, 7] include recall in their metrics and achieve better correlation with human judgements compared with BLEU.

[14] analysed BLEU at the sentence level with Pearson's correlation with human judgements over 1 to 9 grams. In order to apply BLEU for sentence level, they add one to the count of each n-gram. Results shows that BLEU with only uni-gram precision has the highest adequacy correlation (0.87), while adding higher order n-gram precision factors decreases the adequacy correlation and increases fluency. Overall they recommend using up to 5-gram precision to achieve the best balance. [12]'s experiments show that uni-gram and bi-gram precisions contribute over 95% of the overall precision. They also found that adding higher n-gram precision leads to a bias towards fluency over precision. However, it is not clear which of fluency or adequacy is more important, with recent evaluation favouring ranking judgements that implicitly consider both fluency and adequacy [13, 19–21].

These limitations affect the possible applications of BLEU, particularly for MT tuning. In tuning, the references are given, and we want the decoder to produce translations with high BLEU score. Current solutions rank translations in n-best lists [4, 22] or explicitly search for the maximum BLEU translation and use this for discriminative updates [23, 4, 24, 5]. In order to efficiently search for the maximum BLEU translation we need to be able to evaluate BLEU over partial sentences. However, the clipping and high order n-grams make this hard to apply BLEU during decoding. Thus the process relies on coarse approximations.

3 EXPERIMENTS

To address the above mentioned limitations, we analyse each component of BLEU and seek to address these shortcomings. Our prime motivation is to allow for better sentence level evaluation. In what follows, we test the effect of replacing and adjusting each component in BLEU – swapping the precision terms for recall, moving to an arithmetic mean, considering only smaller n-grams, dropping clipping of counts etc. In each instance, we test how each component contributes to BLEU in terms of correlation with human judgement data from previous translation evaluations. Hereinafter we use the following notation to denote each component in our metric:

P n-gram precision
R n-gram recall used in place of precision in Equation 2
F n-gram F-measure used in place of precision, balanced to weight recall 9 times higher than precision
A P/R/F terms are combined using an arithmetic mean
G P/R/F terms are combined using a geometric mean, as in Equation 1
B the brevity penalty term is included
1–4 include P/R/F terms for n-grams up to the given size
C clipping of counts used in P/R/F computation.

Note that our short-hand for standard BLEU is `PGBC4`, while a metric for clipped recall over unigrams and bigrams with no brevity penalty is labelled `RGC2`.

Our experiments are divided in two parts. In the first part we modify BLEU into several variants and compare the evaluation results of variants with human judgements, at both the sentence and document levels. In the second part, BLEU variants are used for parameter tuning, and the system output of each variant is evaluated by human judges. Our baseline BLEU is David Chiang's implementation, and add-1 smoothing is used for sentence level evaluation.

3.1 *Sentence Level evaluation*

For sentence level evaluation we follow the procedure from WMT11 [19], which uses Kendall's tau correlation (equation 3) to measure metrics' quality,

$$\tau = \frac{\text{num concordant pairs - num discordant pairs}}{\text{total pairs}} \tag{3}$$

where two ranked lists of translations according to humans and metrics are compared by counting the number of concordant and discordant relative ordering of pairs of translations, ignoring ties in either human or metric rankings.

We use τ to compare the sentence rankings produced by BLEU and all of our variants against human rankings. The human rankings were collected from WMT 09–11 [21, 20, 19], pooling together the data from

Table 3. Sentence level evaluation results showing τ between the metric-derived rankings and the human rankings. The label in the three columns denotes precision (P), recall (R) or F-measure (F), as used to combine n-gram matches according to each row's metric specification.

	P	**R**	**F**
GBC4	0.2116	0.1942	0.1905
GB4	0.2102	0.1913	0.1868
GC4	0.1879	0.2387	0.2054
ABC4	0.2288	0.2126	0.2076
AB4	0.2267	0.2411	0.2036
AC4	0.2055	**0.2462**	0.2178

Table 4. Results for sentence level evaluation without smoothing counts. Show are Kendall's tau correlations against human rankings. The ^{u} superscript denotes unsmoothed counts and ^{b} denotes smoothed brevity penalty.

	P	**R**	**F**
ABC4^u	0.2351	0.2209	0.2157
GBC4^{u,b}	0.2128	0.1935	0.1900
AC4^u	0.2176	**0.2462**	0.2178

Table 5. Sentence level evaluation results for metrics with various sized n-grams. Results are τ values and bolding shows the best score in each column.

	PGBC	**PGB**	**PABC**^{u}	**RAC**
1-4 grams	0.2116	0.2102	0.2351	0.2462
1-3 grams	0.2252	0.2230	**0.2375**	0.2491
1-2 grams	**0.2295**	**0.2278**	0.2353	0.2501
unigram	0.2284	0.2181	0.2293	**0.2726**

English-Spanish, English-French and English-German, in both translation directions. We selected only sentence pairs that were judged by at least two human annotators and where at least 60% of annotators agreed on their judgements. Our final test set contains 10,278 sentence pairs and has a Kappa of 0.8576.

Tables 3–5 show the results of sentence level evaluation with precision, recall and F-measure. Table 3 shows the results for BLEU variants with add-one smoothing. It is clear that the recall based metrics generally outperform those using precision and F-measure. The best performing metric is the RAC4 variant which combines 1-4-gram recall scores in arithmetic mean with no brevity penalty. This configuration has 3%

higher τ compared to standard BLEU (PGBC4), 0.2462 versus 0.2116. Overall, variants using the arithmetic mean perform better than those using the geometric mean. When clipping is removed, the performance uniformly decreases, but only slightly. More notable is the effect of the brevity penalty. When it is omitted, the performance drops heavily for precision metrics, but increases for recall and F-measure metrics. This is unsurprising as these metrics already disprefer short output. The F-measure based metrics are worse than both precision and recall variants when BP is included, but slightly outperform precision when BP is omitted.

A natural question is how important smoothing of counts is to sentence-level evaluation. Table 4 presents the correlation results for a number of variants.[1] Compared to the smoothed versions in Table 3, the unsmoothed arithmetic mean variants have better performance. We also found that smoothing the brevity penalty, $BP = \exp(1 - \frac{r+\alpha}{c+\alpha})$, using the same value of $\alpha = 1$ gave better performance compared unsmoothed BP.

All the results thus far have used $n = 4$-grams and smaller, following in the footsteps of BLEU. Our next experimental question is revisit this choice and test different values of n. Table 5 shows the sentence-level correlation results for various n-gram sizes, applied to some of the more successful metrics identified above. The most striking result is that RAC1 far exceeds all other metrics, and is one of the simplest in that it only uses unigrams. The arithmetic mean uniformly outperforms the geometric mean (including standard BLEU, PGBC4, in the top left corner). Also interesting is the pattern in the other columns, where the performance is relatively insensitive to the choice of n, with the maximum at $n = 2$ or $n = 3$. Overall the story is clear: large n-grams are not appropriate in this setting, and harm performance.

3.2 *Document Metric Evaluation*

In this section, the performance of BLEU variants will be tested at document level. We follow the WMT08 [25] document level procedure: we compare rankings based on evaluation metrics against human rankings using Spearman's rho correlation, defined as

$$\rho = 1 - \frac{6 \sum_{i=1}^{n} d_i^2}{n(n^2 - 1)} \tag{4}$$

[1] Un-smoothed PBCG4 is not reported as it has very low Kendal's tau correlation.

where d_i measures the difference between the rank value assigned to sentence i by the system versus the human, and the n is number of sentences in the document.

Our test corpora are taken from all systems that were submitted as part of WMT08 for the *test2008* dataset.[2] We selected Spanish, French and German into and out-of-English for our tests. The final score is the average of the BLEU variant Spearman's rho correlation with human ranking in three tasks of *ranking*, *constituent* and *yes/no*. Please see [25] for a full exposition. In brief for the *ranking* and *constituent* the human judges were asked to rank a small set of candidate sentences in order of quality, focusing on a specific syntactic constituent for the latter case, and for *yes/no* they made a binary judgement of acceptability of the translation. Documents level rankings were constructed by counting how often each system outperformed the others, or the ratio of yes to no judgements. For the purpose of our experiments, we present average ρ values over the three different tasks.

Table 6. Document level correlation, measured using ρ.

	PGBC4	RGBC4	PABC4	PGB4	RAC4	PGBC2
es-en	0.7995	0.8111	0.7995	0.7995	**0.8135**	0.7925
fr-en	**0.9501**	0.9267	0.9443	**0.9501**	0.9414	0.9428
de-en	**0.5939**	0.5818	**0.5939**	**0.5939**	**0.5939**	**0.5939**
en-es	0.7757	0.7545	**0.8060**	0.7757	0.7545	**0.8060**
en-fr	**0.9388**	**0.9388**	**0.9388**	**0.9388**	**0.9388**	**0.9388**
en-de	0.7151	0.7151	**0.7212**	0.7151	0.7151	**0.7212**
avg.	0.7955	0.7881	**0.8006**	0.7955	0.7928	0.7992

Table 6 shows the results for document level evaluation, where we have selected promising metrics from the sentence level experiments. All the results are very close together, making it hard to draw concrete conclusions. However we do notice some contrary findings compared to the sentence level results. Most notably, the recall based metric with arithmetic mean (RAC4) performs worse than BLEU (PGBC4). Our earlier finding regarding clipping still holds here, i.e., that it has a negligible

[2] The reason for using a different dataset than for the earlier sentence level evaluation experiments is that only the WMT08 data provides the official document level human ranking results.

difference (compare PGBC4 and PGB4).[3] The overall best performing variant is PABC4, the arithmetic mean using 4-gram precision, brevity penalty and clipping. This metric is very similar to BLEU, simply swapping the geometric mean for the arithmetic mean.

3.3 *Discriminative Training*

Until now we have applied our metrics to human evaluation data, testing whether our variant metrics result in better ranking of MT outputs. However, it remains to be seen whether the metrics might also work effectively as a loss function for tuning a translation system. This is a better test of the metric, as it will encounter a much wider variety of outputs than present in MT evaluation data. For instance, empty sentences, overly long output, output from models with a negatively weighted language model, etc.

In this experiment we investigate parameter tuning of a statistical machine translation system. The system we used for this evaluation is Moses, a phrase-based decoder [3], which we tune using cmert-0.5, David Chiang's implementation of MERT [22]. We use the following (default) features:

- Translation probabilities, including forward & backward lexical probabilities, word count and phrase count.
- Lexicalised distortion model.
- A tri-gram language model, trained on the target side of the parallel corpus.

The training data for this experiment is Europarl-v6 German to English corpus, which is tuned on *dev-newstest2010* from WMT10 [20]. For the test, we use the *de-en* test set from WMT11 [19]. We tuned five different systems, each minimising a different loss function, and then used them to decode the test set. We randomly picked 50 unique output sentences from five systems' outputs for human ranking, asking our judges to rank them best to worst.

The human ranking used in this paper was done on Amazon Mechanical Turk using MAISE [26]. For each ranking judgement, source and reference sentences are provided, and the five candidate sentences are given in random order. The user then decides how to rank the five outputs. We

[3] In further experiments, not reported her, clipping also had little effect on performance for lower orders of n-gram.

repeat each ranking five times with different annotators. Pairwise annotation agreement in this paper is measured by the kappa coefficient [27],

$$\kappa = \frac{P(A) - P(E)}{1 - P(E)} \tag{5}$$

where P(A) is percentage of annotators agree with each other, and P(E) is the probability of agreement by chance, here $P(E) = \frac{1}{3}$. We also measure the self-agreement of each annotator, and discard all data from annotators with low self-agreement. We used 42 annotators and produced a total of 250 rankings, leaving 143 rankings after the self-agreement filtering. The kappa value for the filtered data was $K = 0.40$, with $P(A) = 0.61$.

The results of the human evaluation are shown in Table 7. The key result is that the most consistently good metric from our earlier experiments, PABC4, also did very well here. It outperformed BLEU (PGBC4) in 31% of cases and underperformed in 27% of the cases, for an overall 4% improvement. This improvement is significant with $p < 0.07$, as measured using the paired bootstrap resampling test [28]. Another interesting result is that PGBC2 and PGBC4 have the same performance, i.e., there is no effect of using larger n-grams. Surprisingly BLEU with clipping is only slightly better than the version without clipping (0.29 vs 0.28). We would expect that the unclipped system might systematically over-predict function words, as these will be less heavily penalised, and therefore produce inchorent output (so-called 'gaming' of the metric). However it appears that the larger n-grams stop this degenerate behaviour.

To further analyse the outputs of the various systems, Table 8 shows the various BLEU components of each tuned system's output. The BLEU (PGBC4) tuned system has the highest tri-gram and 4-gram precision and overall BLEU score, but the PGBC2 tuned system output has the highest uni-gram and bi-gram precision, as expected. The recall variant (RGBC4) has the longest sentence length, while omitting clipping had very little effect on sentence length. Overall the differences in BLEU scores are very small, which is surprising given the significant differences in human evaluation results.

4 Conclusions

In this paper we set out to simplify BLEU, revisiting each of the decisions made when it was originally proposed and evaluating the effect on large

Table 7. Results of human evaluations of de→en output from different systems, each trained to optimise a different metric. The values in each cell show how often the system in the column was judged to be better than the system in the row. To see whether a was better than b, one much look at the difference between cells (a, b) and (b, a), i.e., its reflection. Bold values indicate that the system in the column outperformed the system in the row.

	PABC4	PGBC4	PGBC2	PGB4	RGBC4
PABC4	–	0.27	0.26	0.25	0.29
PGBC4	**0.31**	–	**0.29**	0.28	0.28
PGBC2	**0.33**	**0.29**	–	0.21	0.26
PGB4	**0.28**	**0.29**	**0.23**	–	0.24
RGBC4	**0.33**	**0.32**	**0.29**	**0.28**	–

Table 8. A comparison of the BLEU components for the de→en translations produced by MT systems optimising different evaluation metrics, shown as columns. The rows P1-4 denote 1 to 4-gram precision, and LR is the ratio of lengths between system output and the reference, as used in the brevity penalty.

	PABC4	PGBC4	PGBC2	PGB4	RGBC4
P1	0.4684	0.4761	**0.4763**	0.4711	0.4742
P2	0.1659	0.1691	**0.1705**	0.1676	0.1683
P3	0.0811	**0.0824**	0.0807	0.0816	0.0785
P4	0.0369	**0.0388**	0.0367	0.0380	0.0360
LR	1.0043	0.9985	0.9906	0.9985	**1.0072**
BLEU	0.1236	**0.1265**	0.1234	0.1250	0.1226

collections of human annotated MT evaluation data. Our objectives were to allow BLEU to be applied accurately at the sentence level, and pave the way for simpler sub-sentential usage in the future. The experiments turned up a number of interesting results: bi-grams are at least as effective as 4-grams, clipping makes little difference, and recall based metrics often outperform precision based metrics. The most consistent finding was that the arithmetic mean outperforms the geometric mean. Together the findings about clipping and the arithmetic mean augur well for discriminative training, as these together greatly simplify the decomposition of the metric to partial sentences, as required during decoding to find the best scoring hypothesis. Some of the improvements evaporated when moving from human evaluation data to the discriminative training setting, where the models were tuned to optimise each metric. This suggests that human evaluation data in WMT is biased towards similar models (those trained for BLEU), and that it is inherently dangerous to design a metric solely from WMT evaluation data without also evaluating on additional, more varied, data.

Our overall results show an improvement of sentence level correlation to $\tau = 0.2726$ from $\tau = 0.2116$ for sentence-level BLEU, and for a much simpler metric. We therefore recommend that MT researchers consider using one of our simplified metrics in their experiments where single-reference per-sentence application is required. Our intension is to develop a discriminative algorithm to optimise the simplified metric, which will allow for more accurate optimisation while also resulting in higher quality translations.

REFERENCES

1. Papineni, K., Roukos, S., Ward, T., Zhu, W.J.: Bleu: a method for automatic evaluation of machine translation. In: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics. (2002) 311–318
2. Li, Z., Callison-Burch, C., Dyer, C., Ganitkevitch, J., Khudanpur, S., Schwartz, L., Thornton, W.N.G., Weese, J., Zaidan, O.F.: Joshua: an open source toolkit for parsing-based machine translation. In: Proceedings of the Fourth Workshop on Statistical Machine Translation. StatMT '09, Stroudsburg, PA, USA, Association for Computational Linguistics (2009) 135–139
3. Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., Bojar, O., Constantin, A., Herbst, E.: Moses: Open source toolkit for statistical machine translation. In: Annual Meeting of the Association for Computational Linguistics. (2007)
4. Liang, P., Bouchard-Côté, A., Klein, D., Taskar, B.: An end-to-end discriminative approach to machine translation. In: Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics. ACL-44, Stroudsburg, PA, USA, Association for Computational Linguistics (2006) 761–768
5. Chiang, D., Marton, Y., Resnik, P.: Online large-margin training of syntactic and structural translation features. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing. EMNLP '08, Stroudsburg, PA, USA, Association for Computational Linguistics (2008) 224–233
6. Hopkins, M., May, J.: Tuning as ranking. In: Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, Edinburgh, Scotland, UK., Association for Computational Linguistics (July 2011) 1352–1362
7. Song, X., Cohn, T.: Regression and ranking based optimisation for sentence level mt evaluation. In: Proceedings of the Sixth Workshop on Statistical Machine Translation. (2011) 123–129

8. Chiang, D., DeNeefe, S., Chan, Y.S., Ng, H.T.: Decomposability of translation metrics for improved evaluation and efficient algorithms. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing. EMNLP '08 (2008) 610–619
9. Callison-Burch, C., Osborne, M., Koehn, P.: Re-evaluating the role of bleu in machine translation research. In: In EACL. (2006) 249–256
10. Banerjee, S., Lavie, A.: Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. Proceedings of the ACL-05 Workshop (2005)
11. Doddington, G.: Automatic evaluation of machine translation quality using n-gram co-occurrence statistics. In: Proceedings of the second international conference on Human Language Technology Research. HLT '02 (2002) 138–145
12. Zhang, Y., Vogel, S., Waibel, A.: Interpreting bleu/nist scores: How much improvement do we need to have a better system. In: In Proceedings of Proceedings of Language Resources and Evaluation (LREC-2004. (2004) 2051–2054
13. Callison-Burch, C., Koehn, P., Monz, C., Post, M., Soricut, R., Specia, L.: Findings of the 2012 workshop on statistical machine translation. In: Proceedings of the Seventh Workshop on Statistical Machine Translation. (2012) 10–51
14. Lin, C.Y., Och, F.J.: Orange: a method for evaluating automatic evaluation metrics for machine translation. In: Proceedings of the 20th international conference on Computational Linguistics. COLING '04 (2004)
15. Owczarzak, K., Groves, D., Van Genabith, J., Way, A.: Contextual bitext-derived paraphrases in automatic mt evaluation. In: Proceedings of the Workshop on Statistical Machine Translation. StatMT 06 (2006) 86–93
16. Koehn, P., Arun, A., Hoang, H.: Towards better machine translation quality for the german–english language pairs. In: Proceedings of the Third Workshop on Statistical Machine Translation. StatMT '08 (2008) 139–142
17. Hanneman, G., Huber, E., Agarwal, A., Ambati, V., Parlikar, A., Peterson, E., Lavie, A.: Statistical transfer systems for french–english and german–english machine translation. In: Proceedings of the Third Workshop on Statistical Machine Translation. StatMT '08 (2008) 163–166
18. Liu, C., Dahlmeier, D., Ng, H.T.: Tesla: Translation evaluation of sentences with linear-programming-based analysis. In: Proceedings of the Joint Fifth Workshop on Statistical Machine Translation and MetricsMATR. (2010) 354–359
19. Callison-Burch, C., Koehn, P., Monz, C., Zaidan, O.: Findings of the 2011 workshop on statistical machine translation. In: Proceedings of the Sixth Workshop on Statistical Machine Translation. (2011) 22–64
20. Callison-Burch, C., Koehn, P., Monz, C., Peterson, K., Przybocki, M., Zaidan, O.: Findings of the 2010 joint workshop on statistical machine translation and metrics for machine translation. In: Proceedings of the Joint Fifth

Workshop on Statistical Machine Translation and MetricsMATR. (2010) 17–53 Revised August 2010.

21. Callison-Burch, C., Koehn, P., Monz, C., Schroeder, J.: Findings of the 2009 Workshop on Statistical Machine Translation. In: Proceedings of the Fourth Workshop on Statistical Machine Translation. (2009) 1–28
22. Och, F.J.: Minimum error rate training in statistical machine translation. In: Proceedings of the 41st Annual Meeting on Association for Computational Linguistics - Volume 1. ACL '03, Stroudsburg, PA, USA, Association for Computational Linguistics (2003) 160–167
23. Arun, A., Koehn, P.: Online learning methods for discriminative training of phrase based statistical machine translation. In: Proc MT Summit XI. (2007)
24. Tillmann, C., Zhang, T.: A discriminative global training algorithm for statistical mt. In: Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics. ACL-44, Stroudsburg, PA, USA, Association for Computational Linguistics (2006) 721–728
25. Callison-Burch, C., Fordyce, C., Koehn, P., Monz, C., Schroeder, J.: Further meta-evaluation of machine translation. In: Proceedings of the Third Workshop on Statistical Machine Translation. (2008) 70–106
26. Zaidan, O.: Maise: A flexible, configurable, extensible open source package for mass ai system evaluation. In: Proceedings of the Sixth Workshop on Statistical Machine Translation. (2011) 130–134
27. Cohen, J.: A coefficient of agreement for nominal scales. Educational and Psychological Measurement **20:37** (1960)
28. Koehn, P.: Statistical significance tests for machine translation evaluation. In: Proceedings of 2004 EMNLP. (2004)

XINGYI SONG
DEPARTMENT OF COMPUTER SCIENCE,
UNIVERSITY OF SHEFFIELD,
SHEFFIELD, S1 4DP, UK
E-MAIL: <XSONG2@SHEFFIELD.AC.UK>

TREVOR COHN
DEPARTMENT OF COMPUTER SCIENCE,
UNIVERSITY OF SHEFFIELD,
SHEFFIELD, S1 4DP, UK
E-MAIL: <T.COHN@SHEFFIELD.AC.UK>

LUCIA SPECIA
DEPARTMENT OF COMPUTER SCIENCE,
UNIVERSITY OF SHEFFIELD,
SHEFFIELD, S1 4DP, UK
E-MAIL: <L.SPECIA@SHEFFIELD.AC.UK>

IJCLA VOL. 4, NO. 2, JUL-DEC 2013, PP. 45–62
RECEIVED 07/12/12 ACCEPTED 04/03/13 FINAL 06/03/13

Fast Large-Margin Learning for Statistical Machine Translation

GUILLAUME WISNIEWSKI AND FRANÇOIS YVON

Univ. Paris Sud, France

ABSTRACT

Statistical Machine Translation (SMT) can be viewed as a generate-and-select process, where the selection of the best translation is based on multiple numerical features assessing the quality of a translation hypothesis. Training *a SMT system consists in finding the right balance between these features, so as to produce the best possible output, and is usually achieved through Minimum Error Rate Training (MERT). Despite several improvements, training remains one of the most time consuming step in the development of SMT systems and is a major bottleneck for experimentations. Building on recent advances in stochastic optimization and online machine learning, this paper studies a possible alternative to MERT, based on standard and well-understood algorithms. This approach is shown to deliver competitive solutions, at a much faster pace than the standard training machinery.*

1 INTRODUCTION

A statistical machine translation (SMT) system consists of a ruleset and a scoring function. The ruleset, represented either in the phrase table of a phrase-based system or in the rewrite rules of a hierarchical system, generates a set of translation hypotheses for each source sentence. These candidates are then ranked according to a scoring function so designed that the top ranking translation is also the best according to some external quality measure.

In the vast majority of existing SMT systems, the score of a hypothesis is computed as a linear combination of various numerical features.

The vector of coefficients, one for each feature, is learned using a training set made of source sentences and their accompanying translation reference(s), by maximizing some empirical gain over the training set, where the gain, for instance the BLEU score, evaluates the quality of the translation hypotheses obtained for a given weight vector.

Training of a SMT system is made difficult by the form of the inference rule used to compute hypotheses, the typical gains used in MT evaluation that are neither convex nor differentiable and the size of the search space that makes direct optimization intractable. Various heuristic optimization strategies have therefore been put forward, the most successful to date being MERT [1]. In this approach, optimal weights are derived through a complex iterative procedure which repeatedly: *i)* given the weights, decodes the training set to compute an approximation of the search space and *ii)* given this approximated search space, computes an optimal value for the weights.

If MERT has proven to be a practical and effective training procedure, it has been criticized on various grounds, notably for its inability to find good and stable solutions, especially when the feature vector exceeds a dozen dimensions. The computational cost of MERT, due to the need to repeatedly translate the training set, is also viewed as a serious issue: typical runs of MERT can take hours, sometimes days to complete.

Replacing MERT therefore remains a matter of active research. For instance, [2] reports experiments with several variants of MERT, aimed at making its results more stable. Another line of research has been to improve the approximation of the search space, using lists of randomly generated hypotheses [3], word lattices or derivation forests [4]. Inspired by recent advances in structured learning [5], the proposals of [6] and [7] are more radical and replace the gain with training criteria that are easier to optimize. Finally, the recent work of [8] recasts training as a learning-to-rank problem. The main motivation of all these studies was to increase the number of features used during learning, speed being a less important goal.

By contrast, the approach advocated in this work primarily aims at reducing the total training time, which is currently a significant bottleneck for experimentations. Like in [6], an important component of this proposal is the use of a large-margin learning criterion. We depart from existing large margin approaches to SMT by the use of lattices, from which promising pseudo-references (oracles) are efficiently extracted, and the recourse to fast stochastic optimization techniques. The main contribution of this work is to demonstrate, by putting all these ingredients together, that a large scale SMT system can be trained in only a few minutes, the number of decoding passes over the training set being reduced by a factor of almost ten. As discussed below, other advantages of our implementation are its simplicity, especially when compared to [7], and its theoretical guarantees which derive from convex optimization results. As a consequence, our approach does not suffer from stability issues, even for large feature sets.

The rest of the paper is organized as follows. We introduce the large-margin criterion in Section 2 and show how the resulting optimization problem can be easily solved using a subgradient method in Section 3. The optimization procedure is detailed in Section 4. Section 5 presents several MT experiments that show how fast our method is. Related works are summarized in Section 6 and we conclude in Section 7.

2 Large Margin Learning for SMT

2.1 *Notations*

The basic resource for training a SMT system is a training set $D = \{(s_i, r_i)\}_{1 \leq i \leq N}$, made of N source sentences s_i, each accompanied with a reference translation r_i. The set of possible translations for a sentence s_i will be denoted $\mathcal{H}_{s_i} = (\mathbf{h}_{i,j})_{1 \leq j \leq n_i}$. The search space of the decoder is often approximated by an explicit list of n-best hypotheses or by a lattice, which encodes compactly a larger number of potential translations.

Abusing notations, we will denote by $\mathbf{h}_{i,j}$ both a hypothesis (a sequence of words) and its feature representation. Given the search space $\mathcal{H}_{s_i}$ and a weight vector $\mathbf{w}$, translating a sentence s_i thus amounts to solving:

$$\mathbf{h}_i^* = f(s_i; \mathbf{w}) = \operatorname*{arg\,max}_{\mathbf{h} \in \mathcal{H}_{s_i}} \langle \mathbf{h} | \mathbf{w} \rangle \tag{1}$$

where $\mathbf{h}_i^*$ is the predicted translation and $\langle \cdot | \cdot \rangle$ is the dot product in $\mathbb{R}^d$. Using these notations, training a SMT system is the task of finding a weight vector $\mathbf{w}$ such that the predicted translations are as good as possible. Formally, training thus aims to solve the following problem:

$$\mathbf{w}^* = \operatorname*{arg\,max}_{\mathbf{w}} G(D; \mathbf{H}) \tag{2}$$

where the gain function G, for instance the BLEU score, evaluates the quality of the hypotheses $\mathbf{H} = \{\mathbf{h}_i^*, s_i \in D\}$ obtained for a given $\mathbf{w}$.

2.2 *Learning Criterion*

Regularized empirical risk minimization is a popular learning criterion that has proven effective in many applications. Applying it to learn the scoring function of a SMT system amounts to solving:

$$\arg\min_{\mathbf{w}\in\mathbb{R}^d} \frac{\lambda}{2}||\mathbf{w}||^2 + \frac{1}{N}\sum_{i=1}^{N} \ell^{\text{smt}}\left(f(s_i;\mathbf{w}), r_i\right), \tag{3}$$

where $\ell^{\text{smt}}(\mathbf{h}, r)$ is any *sentence level* loss that evaluates the quality of a hypothesis $\mathbf{h}$ with respect to a reference r, (s_i, r_i) is the i-th example, $f(s_i;\mathbf{w})$ is the prediction of the system. The first term of the objective is a regularizer that prevents overfitting and the second is the empirical risk (error on the train set). The hyper-parameter λ controls the strength of the regularization.

Direct optimization of (3) is generally not possible as usual SMT metrics are piecewise constant and therefore not differentiable. However, *structure learning* offers several ways to reformulate this problem in terms of convex programming by deriving upper bounds of arbitrary loss functions thanks to techniques such as margin-rescaling [9] or slack-rescaling [10]. While these upper bounds are not *consistent*, they have achieved optimal prediction accuracy in several tasks. In the following, we will describe the margin-rescaling technique as it can be implemented more easily than slack-rescaling. As detailed in Section 6, the resulting learning criterion is similar to the one optimized by MIRA.

2.3 *Margin Rescaling*

Consider the following generalization of the Hinge loss for the i-th example [9]:

$$\ell_i(\mathbf{w}) = \max_{2\leq j}\left(\ell^{\text{smt}}\left(\mathbf{h}_{i,j}, \mathbf{h}_{i,1}\right) - \langle\mathbf{w}|\mathbf{h}_{i,1} - \mathbf{h}_{i,j}\rangle\right) \tag{4}$$

This loss is convex (as a maximum over a family of linear functions) but is not differentiable everywhere; it is also obviously an upper-bound of $\ell^{\text{smt}}\left(\mathbf{h}_{i,j}, \mathbf{h}_{i,1}\right)$. It results from the following reformulation of the general large-margin classification problem: learning aims at finding a function that scores the correct output $\mathbf{h}_{i,1}$ higher than all other possible outputs $\mathbf{h}_{i,j}$ by a given margin. The worse the prediction of $\mathbf{h}_{i,j}$ compared to $\mathbf{h}_{i,1}$, the larger the margin has to be, which is reflected by scaling the margin by $\ell^{\text{smt}}(\mathbf{h}_{i,1}, \mathbf{h}_{i,j})$ as follows:

$$\langle \mathbf{h}_{i,1} | \mathbf{w} \rangle + \xi_i \geq \langle \mathbf{h}_{i,j} | \mathbf{w} \rangle + \ell^{\text{smt}}(\mathbf{h}_{i,j}, \mathbf{h}_{i,1}) \quad \forall j \geq 2$$

where ξ_i is a slack variable. There are as many constraints as there are possible translations of the source. It is however possible to combine all these linear constraints in a single non-linear constraint:

$$\langle \mathbf{h}_{i,1} | \mathbf{w} \rangle + \xi_i \geq \max_{j \geq 2} \left(\langle \mathbf{h}_{i,j} | \mathbf{w} \rangle + \ell^{\text{smt}} \left(\mathbf{h}_{i,j}, \mathbf{h}_{i,1} \right) \right)$$

Moving the constraints of all examples to the objective of the large margin problem as described in [10] is a simple way to create a convex objective in $\mathbf{w}$ and recover the loss introduced in Equation (4). It must be stressed that, while margin-rescaling (as well as slack-rescaling) offers a generic way to derive a convex upper bound of an arbitrary loss function ℓ, the quality of this bound (how close it is to the "original" loss function) highly depends on the task and the loss function considered.

3 Optimization Procedure

Using the convex upper bound of the evaluation criterion ℓ^{smt} derived in the previous section, large-margin learning for SMT amounts to optimizing:

$$\min_{\mathbf{w}} \frac{\lambda}{2} ||\mathbf{w}||^2 + \frac{1}{n} \sum_{i=1}^{n} \ell_i(\mathbf{w}) \tag{5}$$

where $\ell_i(\mathbf{w})$ is defined in Equation (4).

Several methods have been proposed to solve this optimization problem [9, 10]. Following [11] we propose to solve it using a straight-forward subgradient descent method which can be easily implemented. Subgradient is a generalization of gradient to convex functions that are non-differentiable [12] and can be used in the same way as a gradient to optimize a function.

3.1 *Subgradient Optimization*

One subgradient of the objective (5) is given by:

$$\mathbf{g} = \lambda \cdot \mathbf{w} + \frac{1}{n} \sum_{i=1}^{n} \mathbf{h}_{i,j^*} - \mathbf{h}_{i,1} \tag{6}$$

where:

$$\mathbf{h}_{i,j^*} = \arg\max_{j} \langle \mathbf{h}_{i,j} | \mathbf{w} \rangle + \ell^{\text{smt}} (\mathbf{h}_{i,j}, \mathbf{h}_{i,1}) \tag{7}$$

The expression of $\mathbf{g}$ results from the following properties of a subgradient: *i)* a subgradient is linear; *ii)* if f is differentiable, its only subgradient is the gradient vector itself; *iii)* a subgradient of $\max_y f(x, y)$ is $\nabla_x f(x, y^*)$ for any $y^* \in \arg\max_y f(x, y)$ if f is differentiable with respect to x.

Computing the subgradient related to the i-th example requires solving the so-called *loss-augmented* problem described by Equation (7) and to find the best (oracle) hypothesis $\mathbf{h}_{i,1}$ according the evaluation metric ℓ^{smt}. These two problems are well-defined and, as described in Section 4.2, they can be solved efficiently. As a consequence, implementing this training strategy does not depend on any heuristic design decision, contrary to most existing large margin approaches to SMT.

Subgradient descent can be applied either in a *batch* setting in which parameter updates are performed on the basis of the (sub)gradient information accumulated over the entire training set or in a *online* or *stochastic* setting, in which parameters are updated on the basis of a single example chosen randomly at each iteration. In this case, the expression of $\mathbf{g}$ is simplified as the sum in Equation (6) vanishes.

Even though batch subgradient descent is known to be a slow optimization technique, using it in an online setting leads to fast convergence [13]. That is why, we only considered the online method. However, for stochastic descent, usual methods to find the optimal value of the learning rate, like line search, can not be applied and the learning rate sequence has to be chosen in advance. The optimization procedure is summarized in Algorithm 1.

3.2 *Averaged Stochastic Descent*

While online algorithms can converge to the neighborhood of the optimum very quickly, there are no guarantees that the objective function decreases after each update. Indeed updates are based only on a (noisy) estimate of the true gradient evaluated from a single example and might sometimes point to a wrong direction. This problem is of more importance in subgradient descent as a subgradient is not always a descent direction. That is why, in the learning curves representing the evolution of the objective function with respect to the number of iterations, the value

Algorithm 1: Optimization procedure

```
input : a number of iterations $T$ and a sequence of learning rate $\eta_t$
$\mathbf{w}$= NullVector()
for $t \in [\![1, T]\!]$ do
    pick an example $(s, r)$ randomly
    compute $\mathbf{h}_{i,1} = \arg\max_{\mathbf{h}\in\mathcal{H}_s} \ell^{\text{smt}}(\mathbf{h}, r)$
    compute $\mathbf{h}_{i,j^*}$ according to Equation (7)
    update = $\lambda \cdot w + \mathbf{h}_{i,j^*} - \mathbf{h}_{i,1}$
    w = w - $\eta_t \times$ update
end
```

of the objective function is often observed to wobble around the optimum [14].

One practical way to reduce the fluctuations of the objective function is to *average* the weights over time. Several recent works [15, 16] have shown that *averaged* stochastic gradient descent leads to very fast convergence when the learning rate is set according to their guidelines: in some of their experiments, the optimum is reached after only a single pass over the train set even for large-scale problems.

4 Implementing Subgradient Descent

Implementing the optimization procedure described in the previous section requires us to define a suitable loss function ℓ^{smt} and to efficiently solve both the loss-augmented and the oracle decoding problems. These choices are described below.

4.1 *Loss Function*

Large-margin learning for SMT relies on a loss function ℓ^{smt} to evaluate the quality of a hypothesis with respect to a given reference *at the sentence-level*. Most of the metrics usually used for MT evaluation, such as BLEU or METEOR are computed at the corpus level. Moreover, contrary to these metrics, learning theory assumes that the smaller the loss is, the better the solution, the loss being 0 when the correct answer is predicted.

Several sentence-level approximation of the wide-spread BLEU metric have already been proposed [7], but we used a simpler approximation that enforces the properties of a loss. Our approximation is based on a linear combination of the i-gram precision:

$$\text{score}(\mathbf{h}, r) = \sum_{i=1}^{I} \varXi_i \cdot c_i(\mathbf{h}, r) - \varXi_0 \cdot c_{\text{non}}(\mathbf{h}, r) \tag{8}$$

where $c_i(\mathbf{h}, r)$ is the number of common i-gram in the hypothesis $\mathbf{h}$ and in the reference r, c_{non} is the number of words of the hypothesis that do not appear in the reference and the $\varXi_i$ are positive constants chosen to maximize the correlation between the BLEU score and its approximation.

The score defined by Equation (8) is a compromise between the number of words that the hypothesis and the reference have in common (accounting for the recall) and the number of words of the hypothesis that do not appear in the reference (accounting for the precision). It can be transformed into a loss: $\ell^{\text{smt}}(\mathbf{h}, r) = \alpha - \text{score}(\mathbf{h}, r)$ where α is the score of the best hypothesis. Computing α is needed since our approximation of BLEU is not normalized.

4.2 *Solving the Oracle Decoding and Loss-Augmented Problems*

For a given source sentence, the search space of a SMT system has the form of a directed acyclic graph (a lattice) in which each edge is associated with a phrase and a vector of features describing the cost of emitting this phrase. For simplicity, we assume that there is a single initial state and a single final state. Each path from the initial to the final state in this lattice corresponds to a translation hypothesis ; its feature representation can be worked out by summing the features on the edges and its “output string” by concatenating the phrases of the edges.

Many SMT problems, including the one appearing in Algorithm 1, can be formulated as shortest path problems in a lattice. For instance, the decoding task, described in Equation (1), is the shortest path problem in which the cost of an edge is defined by the opposite of the dot product between the feature representation of edge and the weight vector $\mathbf{w}$. As lattices are acyclic graphs, shortest path problems can be efficiently solved in a time linear in the number of edges and vertices.

Oracle decoding, the task of finding the best hypothesis according to the loss function, can also be performed using a shortest path algorithm, as long as the evaluation metric factorizes in terms of individual edges

[17]. Considering the BLEU-1 approximation introduced in Section 4.1, finding $\mathbf{h}_{i,1}$ amounts to solving:

$$\operatorname*{arg\,min}_{\boldsymbol{\pi} \in \Pi} - \sum_{i=1}^{n} \theta_{\pi_i}$$

where $\boldsymbol{\pi}$ is a path made of m edges $(\pi_i)_{i=1}^{m}$ in the lattice, Π is the set of all paths and θ_{π_i} is the cost of the edge π_i. It is defined by $\theta_{\pi_i} = \Xi_1 \times c_1(w,r) - \Xi_0 \times c_{\text{non}}(w,r)$ where w is the phrase generated by the edge π_i. This approach can be generalized to find oracle hypotheses for higher-order approximation of BLEU score by first transforming the lattice so that each edge generates a n-gram instead of a word. However, for simplicity, we have only considered BLEU-1 approximation in our experiments.

Finally, solving the “loss-augmented” problem of Equation (7) can be done by defining the cost of an edge as the sum of the cost considered by the decoder and the cost considered by the oracle decoder.

In practice, to keep our implementation simple, we chose to rely on an external decoder to produce the lattices: before optimization, the whole training set is decoded using the same initialization as MERT and all the lattices are saved. Preliminary experiments show that this initialization has limited impact as long as the initial values of the weights are not unbalanced (i.e. no weight is set to 0 or a to a large value). Optimization is then performed, as described in Algorithm 1. As for MERT, the lattices can be regenerated occasionally, to make sure that they still represent an accurate approximation of the search space. However, experiments summarized in the next section show that it is sufficient for lattices to be regenerated only once.

An advantage of this implementation is that it can be used with *any* SMT system. Another way to proceed would be to decode and generate the lattice for sample s_i on an as-needed basis, i.e. upon updating the parameter value based on this particular sentence. While this solution might hasten convergence, it would require a tighter integration with the decoder and also more engineering work to avoid launching the decoder for each example.

5 Experiments

We now describe the experiments made to validate our approach. Recall that our main motivation is to provide a much faster in-place replacement of MERT: we are mainly interested in learning time and have only considered small and standard feature sets.

5.1 *Experimental Setup*

Two data sets were considered: the TED-talk English to French data set provided by the IWSLT'11 evaluation campaign and the French to English Europarl data set provided by the WMT'11 campaign. In all our experiments, we used the Moses decoder.

The TED-talk data set is a small data set made of a monolingual corpus ($111,431$ sentences) used to train the language model and a bilingual corpus ($107,268$ sentences) used to extract the phrase table. The Europarl system is trained using the parallel EPPS and News Commentary corpora ($1,940,639$ sentences). The target side of these two corpora were used to estimate a 4-gram language model with KN-smoothing.

For the TED-talk task, we used `dev-2010` dataset for training and `test-2010` for evaluation; the Europarl system was tuned on the dataset `test-2009` and evaluated on `test-2010`. Training for TED-talk task took 11 decodings of the training set (a wall time of almost 4 hours[1]) of the training set and achieved a BLEU score of 26.12 on the training set an of 23.28 on the test set; for Europarl, training took 10 decodings (more than 6 hours) and achieved a BLEU score of 21.47 on the training set and of 21.10 on the test set.

All reported BLEU scores were computed using the `multi-bleu` tool provided by Moses.As explained above, lattices are regenerated only once, after 300 iterations. Results fall down by about 2 BLEU points when the lattice are not regenerated, but regenerating the lattices more often did not yield any improvement.

5.2 *Learning Speed*

We first analyze the performance of the optimization procedure introduced in Section 3 by studying the evolution of the structured loss during optimization. Recall that the structured loss is a convex upper bound of (an approximation of) the BLEU score which defines the objective function optimized during training.

[1] All experiments are run on a single core of a server with 64G of RAM and 2 Xeon CPUs with 4 cores at 2.3 GHz. All reported times are wall time and include data loading.

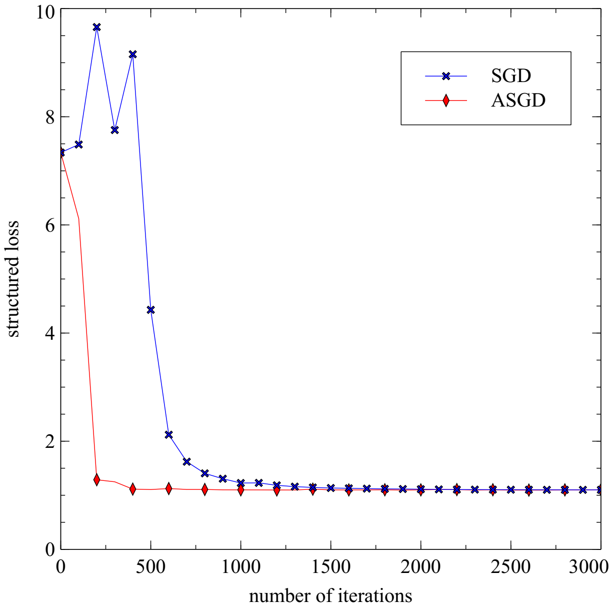


Fig. 1. Convergence of the (sub)gradient descent: evolution of the loss on TED-talk training set

Figure 1 represents the structured loss on the train set of the TED-talk task for two optimization strategies: plain stochastic gradient descent (SGD) and averaged stochastic gradient descent (ASGD). In both case, the learning rate has been set, according to the recommendations of [15]. It clearly appears that the neighborhood of the optimum is reached very quickly: for ASGD, Algorithm 1 converges after having seen only a few hundred examples. However, after reaching the optimum neighborhood, the weight vector is still changing and the objective function continues to decrease, albeit very slowly: the difference between two successive values after $1,000$ iterations is still in the order of 10^{-3}, which is much larger than the stopping criteria that are usually used. A difference in the order of 10^{-6} is only reached after $6,000$ iterations. Similar observations were made on the Europarl data.

To understand why convergence is fast, we have represented in Figure 2 the cosine similarity between the gradients of two examples of the training set after the first iteration of Algorithm 1. It appears that most

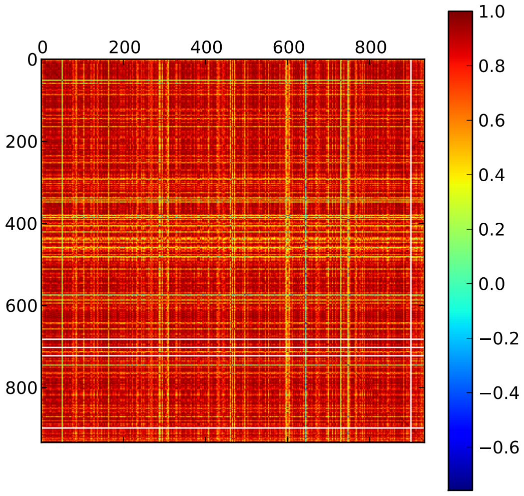


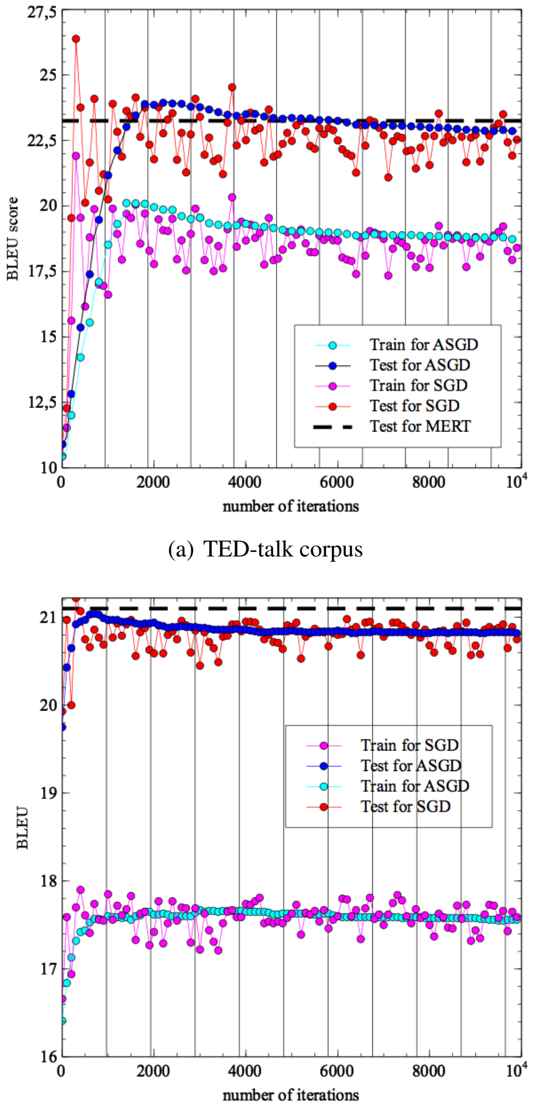
Fig. 2. Cosine similarity between the gradient of the examples in the TED-talk training set (most pairs show high similarity; lighter areas correspond to values in the middle of the scale)

gradients are very similar. This implies that the update in the online setting (based on a single example) is close to the update in the batch setting (after all examples have been seen), and that an online update, which requires N times less computation than a classical gradient update, will give (almost) the same results.

5.3 *Evolution of the BLEU Score*

As shown in the previous section, our optimization method is able to find the optimum of the learning criterion very quickly. However, this criterion is only an approximation of the BLEU score used to evaluate translation quality. In this section we study the quality of this approximation.

Figure 3 represents the evolution of the BLEU score on our two corpora. For SGD, the BLEU score on both the training set and the test set keeps changing during optimization: on the TED-talk training set, after $1,000$ iterations, the amplitude of the variations is still of several BLEU points even though the structured loss is almost stabilized. For ASGD,



(a) TED-talk corpus

(b) Europarl corpus

Fig. 3. BLEU scores on TED-talk and Europarl corpora. The dashed horizontal lines correspond to the score on the test set achieved by MERT and the vertical lines indicate iterations proportional to train set sizes.

Table 1. Comparison of MERT and of our approach (using binarized models).

	method	BLEU	# decodings of training set	training time (+ time to generate lattices)
TED-talk	MERT	23.28	11	3h39
	online	23.98	1.3	3mn (+ 5mn25)
Europarl	MERT	21.10	10	5h25
	online	21.04	1.3	6mn34 (+ 7mn30)

the regularization of the weight vector that results from its averaging over time reduces significantly the fluctuations of the BLEU scores. Nevertheless, for the two tasks, the trend is the same: at the beginning, performance quickly improves during the first few hundred iterations and then decreases slowly. Also note that *i)* the lattices have been regenerated only once during the optimization and that *ii)* the optimum BLEU value is reached, depending on the task, after $1,000$ or $2,000$ iterations. The corresponding total learning time is less than a few minutes with our simple and non-optimized implementation in Python. Table 1 summarizes the performances achieved by our approach and traditional MERT training.

In both cases, the observed variations of BLEU indicate that the upper bound used during optimization is not tight, which results from one of the following reasons: *i)* the way the optimization problem is convexified, *ii)* our sentence-level approximation of BLEU or *iii)* the additional approximations made when solving the loss-augmented or oracle decoding problems. To find out the source of the observed discrepancy, we have represented, in Figure 4, the evolution of both the BLEU-4 score used to evaluate translation quality and the average over the whole training set of the sentence-level BLEU-1 used during optimization. While these two scores are initially correlated (both of them are steadily increasing), this correlation seems to weaken with the number of iterations and, at the end, the BLEU-4 score is decreasing even if the BLEU-1 approximation continues to grow. Further experiments are still required to understand if this problem is the only responsible for the evolution of the BLEU-4 score during optimization.

5.4 *Stopping Criterion*

As shown in Figure 3, upon converging, our learning method is slightly outperformed by traditional MERT training. However, some of the weight

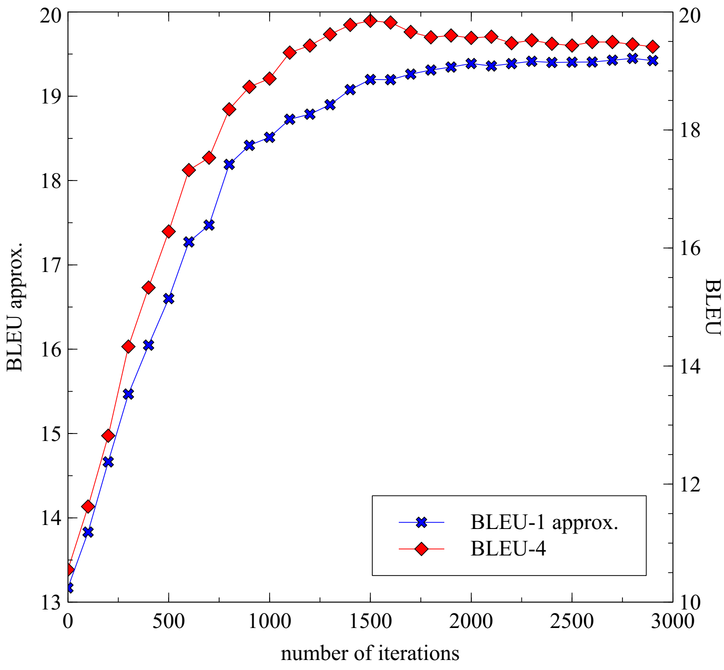


Fig. 4. BLEU-4 and BLEU-1 approximation during optimization on TED-talk for ASGD.

vectors found during optimization achieved better performance. Our approach is, therefore, only useful if we can find a criteria for stopping the optimization when a "good" weight vector is found. Fortunately, in all our experiments, we found that the BLEU scores on the training and on the test sets are highly correlated: their Pearson correlation coefficient is more than 0.92. The point that achieves an optimal BLEU score on the test set can therefore be easily identified by computing BLEU scores on the training set, which is done efficiently using a shortest path algorithm in the lattices without decoding the data again.

For the TED-talk task, the best point found by this method and the ASGD strategy slightly outperforms MERT by 0.7 points, while on the Europarl task MERT is better by 0.06 points. Using the SGD strategy leads to larger improvements at the expense of a higher variability in the score on the test set.

6 RELATED WORK

This paper is inspired by recent works on using structure learning techniques for SMT. This trend was pioneered by [18], who proposed to use a structured perceptron to train a PBSMT system. Like [6, 7], our approach augments the simple perceptron loss with a margin term. We however depart from these implementations in several ways. A first important difference is the use of an alternative optimization strategy, which, contrarily to the existing implementations of MIRA for MT, is really online and updates parameters after processing each instance. This is motivated by the observations of Section 5.2 and significantly speeds up learning. Another important difference is the use of lattices..

There are a number of additional small differences from MIRA, such as the approximation of the BLEU score, and the specific choice of the pseudo-reference: while the policy advocated in [7] selects a hypothesis that has both a high BLEU score and a good model score, our approach simply looks at BLEU scores. Incidentally, this difference makes our loss function slightly different from the one used in [7], as our pseudo-references are less dependent on the current value of the parameters. Altogether, it seems fair to state that our approach is conceptually much simpler to understand, to implement and to reproduce than approaches inspired by MIRA, which rely on the setting of many parameters such as the size of the n-best list, the slack parameter, the selection strategy for oracle hypotheses and their number, etc.

7 CONCLUSION

Building on recent advances in stochastic optimization and online machine learning, we have presented in this work an optimization method for the training of SMT systems. Our method achieved results that are at least as good as traditional MERT training, while being much faster. Another advantage of this technique is that it is based on the optimization of a convex objective function, implying that the resulting optimum will be less subject to variations, even in the presence of large feature sets.

While the performance obtained with a simple and straightforward implementation are already good, several questions remain open. We are, in particular, interested in understanding the impact of lattice sizes and of considering more features. Our future work will include a truly online implementation of this learning method within an open source decoder as well as a head to head comparison with MIRA.

ACKNOWLEDGMENTS This work has been partly financed by OSEO, the French State Agency for Innovation, under the Quaero program.

REFERENCES

1. Och, F.J.: Minimum error rate training in SMT. In: Proc. ACL'03, Sapporo, Japan (2003) 160–167
2. Foster, G., Kuhn, R.: Stabilizing minimum error rate training. In: Proc. WMT, Athens, Greece (2009) 242–249
3. Chatterjee, S., Cancedda, N.: Minimum error rate training by sampling the translation lattice. In: EMNLP'10, Stroudsburg, PA, USA, ACL (2010) 606–615
4. Kumar, S., Macherey, W., Dyer, C., Och, F.: Efficient minimum error rate training and minimum bayes-risk decoding for translation hypergraphs and lattices. In: Proc. ACL'09. (2009) 163–171
5. Smith, N.A.: Linguistic Structure Prediction. Synthesis Lectures on Human Language Technologies. Morgan and Claypool (May 2011)
6. Watanabe, T., Suzuki, J., Tsukada, H., Isozaki, H.: Online large-margin training for statistical machine translation. In: Proc. EMNLP'07, Prague, Czech Republic (June 2007) 764–773
7. Chiang, D., Marton, Y., Resnik, P.: Online large-margin training of syntactic and structural translation features. In: EMNLP'08. (2008)
8. Hopkins, M., May, J.: Tuning as ranking. In: EMNLP'11, Edinburgh, Scotland, UK., ACL (July 2011) 1352–1362
9. Taskar, B., Guestrin, C., Koller, D.: Max-margin Markov networks. In: NIPS 16. MIT Press, Cambridge, MA (2004)
10. Tsochantaridis, I., Joachims, T., Hofmann, T., Altun, Y.: Large margin methods for structured and interdependent output variables. JMLR **6** (December 2005) 1453–1484
11. Ratliff, N., Bagnell, J.A., Zinkevich, M.: (online) subgradient methods for structured prediction. In: Artificial Intelligence and Statistics. (2007)
12. Shor, N.Z.: Minimization Methods for Non-differentiable Functions. Springer-Verlag (1985)
13. Bertsekas, D.P. In: Incremental Gradient, Subgradient, and Proximal Methods for Convex Optimization. MIT Press (2012) 85–119
14. Bottou, L.: Online algorithms and stochastic approximations. In Saad, D., ed.: Online Learning and Neural Networks. Cambridge University Press, Cambridge, UK (1998)
15. Xu, W.: Towards optimal one pass large scale learning with averaged stochastic gradient descent. CoRR **abs/1107.2490** (2011)
16. Bach, F., Moulines, E.: Non-asymptotic analysis of stochastic approximation algorithms for machine learning. In: NIPS 24. (2011)

17. Sokolov, A., Wisniewski, G., Yvon, F.: Computing lattice bleu oracle scores for machine translation. In: EACL'12, Avignon, France, ACL (April 2012) 120–129
18. Liang, P., Bouchard-Côté, A., Klein, D., Taskar, B.: An end-to-end discriminative approach to machine translation. In: ACL, Sydney, Australia (2006) 761–768

GUILLAUME WISNIEWSKI
LIMSI—CNRS,
UNIV. PARIS SUD,
91403 ORSAY CEDEX, FRANCE
E-MAIL: <WISNIEWS@LIMSI.FR>

FRANÇOIS YVON
LIMSI—CNRS,
UNIV. PARIS SUD,
91403 ORSAY CEDEX, FRANCE
E-MAIL: <YVON@LIMSI.FR>

IJCLA VOL. 4, NO. 2, JUL-DEC 2013, PP. 63–78
RECEIVED 07/12/12 ACCEPTED 04/03/13 FINAL 14/06/13

Syntax-Based Reordering in Phrase-Based English–Hungarian Statistical Machine Translation

LÁSZLÓ J. LAKI, ATTILA NOVÁK, AND BORBÁLA SIKLÓSI

Pázmány Péter Catholic University, Hungary

ABSTRACT

Phrase-based statistical machine translation systems can generate quite high quality translations in the case of language pairs with similar structure and word order. However if the languages are more distant from a grammatical point of view, the quality of translations is much behind the expectations, since the baseline translation system cannot cope with long distance reordering of words and morphological synchronization. In our paper, we present a method that tries to overcome these problems in the case of English-to-Hungarian translation. We describe how we defined some reordering rules on the English sentences in order to approximate the syntax of a hypothesized Hungarian translation prior to the actual process of translation. Due to the limited training corpus and data sparseness, and problems caused by the agglutinating characteristics of Hungarian, we applied a morpheme-based translation system. We show that although automatic evaluation cannot reliably reflect the improvement, human evaluation of the systems shows that readability and grammatical correctness of the translations were improved.

KEYWORDS: *Statistical machine translation, morphology, reordering.*

1 Introduction

Currently, the most widespread method for machine translation is to train statistical machine translation (SMT) systems without much explicit specific knowledge of the actual language pair, instead of creating sophisticated language dependent rule-based systems. For syntactically similar and morphologically simple language pairs, methods of phrase-based SMT perform quite well. However, in the case of more distant languages (such as English and Hungarian), there are less promising results. Studies have also shown that increasing the size of the training corpus still does not provide significant increase in the quality of translation [1]. Due to free word order and rich variability of word forms in Hungarian, even big corpora represent grammatical phenomena very sparsely. It implies that SMT systems applied for the English-Hungarian language pair are compromised by data sparseness problems. Our goal was to create a hybrid translation system that, while exploiting the advantages of statistical methods, tries to decrease the above mentioned difficulties.

2 Machine Translation from English to Hungarian

2.1 *Characteristics of Hungarian*

Hungarian is an agglutinating and compounding language with a practically unlimited number of different word forms. This, combined with free word order of main grammatical constituents and systematically different word order in NP's and PP's, results in a poor performance of simple phrase-based English to Hungarian translation systems. The great number of mismatches in word order and word count, the frequent need of long distance word movement and the low representing power of unanalyzed corpora for an agglutinating language like Hungarian, are all factors that make English-to-Hungarian machine translation difficult. The following comparison of language-specific corpus characteristics illustrates the latter problem. While the number of different word tokens in a 10 million word English corpus is generally below 100,000, it is well above 800,000 in the case of a Hungarian corpus of the same size. However, the 1:8 ratio does not correspond to the ratio of the number of possible word forms between the two languages: while there are no more than about 4–5 different inflected forms for an English word, there are about a 1000 for a Hungarian word, which indicates that a corpus of the same size is much less representative for Hungarian than it is for English [2].

2.2 *SMT and Word Order Differences*

If we evaluate the performance of phrase-based machine translation systems between English and various other European languages, we find that these systems perform much worse for languages which differ significantly from English in terms of word order. This indicates that the generic reordering algorithms implemented in phrase-based SMT systems cannot handle long distance word order mismatches effectively. In this paper, we describe a system that uses language-pair-dependent movement rules to handle word order differences, which were implemented as pre- and postprocessing steps around the core of a phrase-based SMT system.

3 Applying Reordering Rules

In order to reduce the complexity of the translation task, our system applies reordering rules prior to training the statistical models. The transformations applied to the source sentences make them more similar to the structure of the corresponding target sentences. In order to perform the required word movements, the rules rely on constituent structure and typed dependency relations in the English source sentences. To process raw sentences, the Stanford parser [3] is used as described in Section 4.2. This enrichment of the grammatical description of the sentence provides enough information for defining rules that can transform the source sentence structures to others that correspond to those occurring in the corresponding hypothesized Hungarian sentence. Since the SMT system is based on data extracted from aligned phrases in the parallel training corpus, the quality of the alignment phase is of crucial importance [4]. Thus one of our goals for the reordering rules was to create a better source for the alignment module. We expected that training the system on such a set of transformed English–Hungarian parallel sentences, more representative statistics can be built than in the case of the baseline model.

Approximating the structure of the source and target languages to each other can on the one hand decrease word alignment errors that result from differences in the organization of morphemes to surface word forms. On the other hand, results published on the research of other language pairs (such as English–German or English–Turkish) have shown that by applying reordering rules to the source sentence, the number of words left without translation during decoding can be decreased [5–7].

We created rules only for those word order differences which are systematically present between the two grammars: e.g. prepositions vs. case endings/ postpositions, possessive determiners vs. possessive suffixes etc. We did not intend to handle free word order variations of Hungarian, where the same meaning can be expressed with several different orderings, since in Hungarian, the actual word order in a sentence is not only determined by syntactic, but also by pragmatic factors.

Reordering rules rely both on phrase structure and dependency relations in the English input sentences. Once having these relations extracted, transformations are carried out along the relevant relations. A simple example is a phrase like *in my house*, which is transformed to the form *house_my_in* corresponding to the single word *házamban* in Hungarian. The morphological segmentation of this word is *ház[N] + am[PxS1] + ban[Ine]*, with the Hungarian morphemes corresponding to 'house[Noun] + my[Possessor:1Sg] + in[Case:Inessive]'.

Defining and applying the rules for such short phrases is not particularly difficult. However, related words in longer sentences can be much further separated from each other and they may be involved in more than one relation which often results in an interaction of word order constraints. In a similar manner, some rules insert morphological elements corresponding to those that are present in the Hungarian sentence, but not explicitly expressed in English, such as the accusative case suffix. These morphemes are important for the accuracy and fluency of the translation.

Our reordering rules fall into three categories:

3.1 *Rules Affecting Word Order and Morpheme Division/Unification*

Once having the dependency relations extracted from the sentence, these rules are responsible for moving each word to its reordered position and at the same time performing unification of English function words in order to make English sentence structures more similar to Hungarian. Besides typed dependencies, these transformations also rely on the constituent parsing of the sentences. Some examples of these rules are the ones transforming passives, positioning auxiliaries, prepositions and transforming possessive phrases. The order of performing these rules is important, especially when longer sequences are affected. In the following sentence in Table 1, we perform two transformations.

While heavy participle phrases in English generally follow the NP they modify, this is never the case in Hungarian where modifiers containing participles strictly precede the noun just like ordinary adjectival modifiers. Moreover, any arguments or adjuncts of the participle must precede it (unlike in the corresponding English structure where they fol-

Table 1. An example of reordering and word form restructuring

Original sentence:	The/DT sons/NNS of/IN the/DT many/JJ merchants/NNS living/VBG in/IN the/DT city/NN ./.
Reordered sentence:	the/DT city/NN_in/IN living/VBG many/JJ merchants/NNS sons/NNS_of/IN ./.

low it). This is an example of a systematic word order difference between the two languages. Correspondingly, the prepositional phrase *living in the city* is transformed along the relations PARTMOD(merchant, living)[1], PREP(living, in)[1] and POBJ(in, city)[1]. First the preposition is attached to the child of the POBJ relation (the head of the dependent NP), then this unified word is moved before the participle and the whole participial modifier phrase before the head noun. Thus the resulting word forms and their order is corresponding to the Hungarian translation: *a város_ban élő* ('the city_in living'). The other phrase (*the sons of the merchants*) is transformed similarly to the resulting *merchants sons_of* order, which corresponds to the order of morphemes in the Hungarian translation of the phrase: *kereskedők fi_ai*.

Table 2. Examples of reordering and morpheme insertion

Original sentence:	That/DT is/VBZ the/DT account/NN at/IN the/DT largest/JJS bank/NN in/IN Bern/NNP ./. ”/”
Reordered sentence:	That/DT is/VBZ the/DT Bern/NNP_in/IN xxx/xxx largest/JJS bank/NN_at/IN xxx/xxx account/NN ./. ”/”
Original sentence:	Buckets/NNS containing/VBG milk/NN must/MD be/VB covered/VBN
Reordered sentence:	Milk/NN_acc/ACC containing/VBG Buckets/NNS must/MD covered/VBN_MD_they/P3

Although in most cases the English sentence has more words than the corresponding Hungarian sentence since English grammatical words usually correspond to bound morphemes in Hungarian, there are situations where some words are missing and have to be inserted in order to get the Hungarian sentence structure. One construction where this hap-

[1] PARTMOD=participal modifier, PREP=prepositional modifier, POBJ=object of preposition. The full list of dependency relations can be found in `http://nlp.stanford.edu/software/dependencies_manual.pdf`

pens is the case of postnominal modifiers not containing a participle (e.g. *the largest bank in Bern*) which are transformed into prenominal modifiers in Hungarian that do contain one. Since the participle to be inserted depends on the context, we insert only an abstract character string representing the participle, the actual realization of which is determined by the SMT system during translation based on similar transformed examples in the training corpus. One such example is the sentence in Table 2 containing the string *xxx/xxx* that is translated to Hungarian as *levő* ’being’. The other example in Table 2 shows insertion of the accusative ending in addition to movement and reordering of the participle modifier that contains it.

3.2 *Rules Affecting Only Morphological Structure, Not Word Order*

English sentences contain several types of implicit structural information that are represented as explicit suffixes in Hungarian. E.g., while objects are identified by their position in English, the same dependency relation is explicitly marked by the accusative case suffix *-t* in Hungarian. Since dependency parsing identifies the object relation in English, it can be transferred as an additional morpheme to the reordered sentence. For example, the original sentence *She/PRP shot/VBD herself/PRP ./.* is transformed into the sentence *shoot/VB_Past_she/PRP herself/PRP_acc/ACC ./.*

There are cases when English represents some morphemes as separate words, while these are only suffixes in Hungarian. To avoid the aligner connecting these morphemes to some other words on the Hungarian side, these words are attached to their corresponding position. For example, if the sentence contains a possessive determiner and the object of the possession, then these are connected. Thus the phrase *”my/PRP$ own/JJ mother/NN”* is transformed to the form *”own/JJ mother/NN_my/PRP$”*, which corresponds to the Hungarian phrase *”saját anyá_m”*.

3.3 *Minor Adjustment Rules*

Rules in this group make some adjustments necessary to make the results of previous transformations well-formed. E.g., the transformations produce two consecutive definite articles if the possessor and the possessed are both definite in a possessive construction or if a definite noun has a modifier that contains another definite dependent. E.g., the phrase *the house standing in the forest* would be transformed to *the ∗the forest_in standing house*. Only one definite article is present in Hungarian in constructions of this kind: the extra articles are deleted by a minor adjustment rule. We also classified some simple movement rules as minor adjustment rules, as these do not interact with others in a complicated manner. One example is the attachment of the genitive *'s* (see Table 3) or the transposition of currency symbols after the sum they belong to.

Table 3. An example of possessive reordering

Original sentence:	John's cat

Dependency relations:	poss(cat, John)
	possessive(John, 's)

Reordered sentence:	John/NNP cat/NN_'s/PoS
Hungarian sentence:	John macská_ja

4 Tools and Resources

4.1 *Corpora*

The available English–Hungarian corpora are usually not suitable for training a general purpose SMT system, since they contain the terminology of a certain specific domain. That is why we used the largest and thematically most general corpus, called Hunglish[8], created by BME MOKK[1] and the Research Institute for Linguistics of the Hungarian Academy of Sciences. This corpus contains parallel texts from the following domains: literature and magazines, law and movie subtitles. There is a great degree of variation in the quality of different parts of the corpus. We automatically eliminated sentence pairs from the corpus that caused technical problems, but overall translation quality was not checked. Finally, the number of sentence pairs we used for training the system was 1,202,205 parallel sentences, which is 12,396,277 words on the English side and 12,316,157 on the Hungarian side.

[1] MOKK Centre for Media Research and Education at the Department of Sociology and Communication, Budapest University of Technology and Economics

4.2 *Constituent and Dependency Parsing*

For the first step of preprocessing, the English sentences were parsed, and dependency relations were extracted. To perform a morpheme-based translation, a part-of-speech tagger was also necessary for Hungarian.

To annotate the Hungarian side of the corpus, we used the PurePos automated morphological annotation system [9]. We parsed the Hungarian side of the corpus using this tool decomposing morphologically complex words in order to have a denser representation of the corpus than the unanalyzed version containing only word forms.

Since the original surface word forms can be reconstructed from the lemma and the morphological tags, the statistics for word alignment and translation can be improved by considering only the lemmas, as they occur more frequently in the corpus than any of the inflected forms. By applying this methodology, the translations generated by the SMT system also contain sequences of lemmas and morphosyntactic tags, thus in order to generate the final form of the translated sentence, the surface form of the words have to be regenerated. We did this by applying the word form generator module of Humor morphological analyzer to the output of the decoder [10, 11].

For parsing English, we used the state-of-the-art Stanford parser [3]. Since the quality of syntactic analysis is a crucial factor for reordering, we used the slower, but better lexicalized version of the parser. This results in a bit more accurate parses than the baseline unlexicalized parser, but it still very frequently generates parses which are often agrammatical with agreement errors and odd PoS sequences like the ones in Table 4.

Table 4. Examples of low level errors affecting reordering

POS-tagged sentence	-/: 100/CD million/CD **sound/NN** good/JJ to/TO me/PRP ./.
Reordered sentence	-/: me/PRP_to/TO xxx/xxx 100/CD million/CD sound/NN good/JJ ./.
POS-tagged sentence	For/IN airline/NN personnel/NNS ,/, we/PRP **cash/NN** personal/JJ **checks/VBZ** up/RP to/TO $/$ 100/CD ./.
Reordered sentence	airline/NN personnel/NNS_For/IN ,/, cash/NN personal/JJ up/RP_checks/VBZ_we/PRP 100/CD_$/$_to/TO ./.

Due to the sequentially pipelined construction of the system, errors are propagated from the very first PoS tagging step through the whole transformation and translation process. Each component of the pipeline assumes correct input, which they do not try to correct. Rather, they try their best to accommodate to whatever input they receive, often resulting in an absurd output. Word and phrase misplacements due to these wrong analyses yield a critical source of errors in the whole system, since the reordering rules are executed on erroneous input. It means that if we reorder an erroneously parsed sentence, then it is likely that the reorderings worsen the final result of the translation rather than improving it. The first such source of error is wrong PoS tag assignment. The most typical error is confusing nouns, adjectives and verbs, which is usually of fatal consequences regarding the translation of the sentence. Since both constituency and dependency parsing are based on such misleading information, the error propagates resulting in mistakes such as the ones displayed in Table 4.

5 The Moses Toolkit

In our present work, we used the phrase-based Moses SMT toolkit [12] to perform our translation experiments. Moses is the most widely used SMT tool. It is a practical solution for the tasks of both training and decoding. It depends on several external tools for the creation of the language models and the evaluation of the system.

The Moses system is suitable for implementing a so-called factored translation system. Instead of relying on just the surface form of the words, further annotations, such as morphological analysis, can be used in the process of a factored translation. Translation factors might be the surface form of each word, its lemma, its main PoS tag, its morphosyntactic features. During factored translation, there is an opportunity to use multiple translation models, generation models or contextual language models. Since the system has the possibility to use any combination of these, in theory, it is able to generate better translations using sparse linguistic data than a word-based baseline system. This feature is vital in cases where some abstraction is necessary, because some words in the sentence to be translated or generated are missing form the training set.

We investigated both factored and morpheme-based translation as possibilities to cope with data sparseness problems when translating from English to Hungarian. However, we found that traditional factored training and decoding is not suitable to handle the massive data sparseness issues encountered when translating to aggutinating languages like Hungarian or Finnish (see e.g. [13] for similar conclusions for the applicability of factored models to translation to Finnish). Nevertheless, factored models may be applicable to the solution of certain problems and are subject of our further investigation. The baseline system that we used for comparison is trained on the raw corpus without any preprocessing.

5.1 *Morpheme-based Translation*

In the morpheme-based implementation, morphological analysis, parsing and the reordering rules were applied to the corpus before training and translation, but at the end, no generation of word forms were performed within the Moses framework: the output of the decoder is a sequence of morphemes. We performed an automatic evaluation of this morpheme-based translation output using the BLEU metric. In contrast to the traditional surface-word-form-based BLEU score (w-BLEU), this score, which we term mm-BLEU, is based on counts of identical abstract morpheme sequences in the generated and the reference translations instead of identical word sequences. Note that this also differs from m-BLEU as used e.g. in [13], which is BLEU applied to (pseudo-)morphs generated by an unsupervised segmenter. mm-BLEU represents the ability of the system to generate the correct morphemes in the translations. After having these morphemes translated, a morphological generator was applied to the output of the Moses decoder in order to acquire the final word forms. As shown in Table 5, this resulted in lower w-BLEU scores than that of the baseline system. Nevertheless, manual investigation of the translation outputs revealed that the morpheme-based system is better at capturing grammatical relations in the original text and rendering them in the translation by generating the appropriate inflected forms. Although it is not reflected by the w-BLEU scores, it generates better translations from the perspective of human readability than the baseline system.

6 RESULTS

Since human evaluation is slow and expensive, machine translation systems are usually evaluated by automated metrics. However, it has been shown that system rankings based on single-reference BLEU scores often do not correspond to how humans evaluate the translations, for this reason, automatic evaluation has for a long time not been used to officially rank systems at Workshops on Statistical Machine Translation

(WMT) [14]. In our work, we present results of automated evaluation using a single reference BLEU metrics, but we also investigated translations generated by each system using human evaluation applying the ranking scheme used at WMT workshops to officially rank systems.

Our experimental setting for automated evaluation consisted of three separate test sets of 1000 sentences each, which were separated from our corpus prior to training the system. Besides these, evaluation was performed on a test set of a different domain (news) that is not represented in the training set at all.

Table 5 contains the traditional word-based w-BLEU scores of the baseline, the morpheme-based mm-BLEU scores of the morpheme-based system with rule-based reordering and w-BLEU scores of the latter system with the target language surface word forms generated. The w-BLEU scores are lower compared to the baseline for all the test sets. However, as mentioned above, the decrease in these values does not necessarily correspond to worse translations.

It is also worth mentioning that morpheme-based mm-BLEU scores for the out of domain newswire test corpora is as high as for the in domain test sets, while the w-BLEU scores are significantly lower for the news test sets.

Table 5. BLEU scores of the word-based baseline and the reordered morpheme-based system

Name	Baseline	Reordered morph.-based	
	w-BLEU	mm-BLEU	w-BLEU
test1	15.82%	64.14%	12.61%
test2	14.60%	57.39%	13.95%
test3	15.04%	57.84%	12.98%
news2008	6.45%	59.73%	6.99%
news2009	7.36%	60.56%	7.26%

During the evaluation process, translations are compared to a single reference sentence. Thus if the machine translation result contains an absolutely wrong word or word form, the evaluation will be just as bad as if it contained a synonym of the correct word, or just a slightly different inflected form of it. The measurements clearly reflect however that translating a test set of different style and domain than the training set, results in much lower BLEU score.

6.1 *Human Evaluation*

We randomly selected a set of 50 sentences from test set 1 that underwent human evaluation as well. Four annotators evaluated translations generated by each of the three systems plus the reference translation in the corpus with regard to translation quality (considering both adequacy and fluency in a single quality ranking). The order of translations was randomized for each sentence. The systems were ranked based on a score that was defined as the number of times a system was found not worse than the other in pairwise comparisons divided by the number of pairwise comparisons. The aggregate results of human evaluation are listed in Table 6.

Table 6. Human evaluation including the reference translations

Name	Baseline	Morph.-based	Reference
test1	34.08%	**52.49**%	83.08%

The ranking produced by each annotator was identical. The rather low score (83.08%) for the reference translations indicates that there are quite serious quality problems with the corpus (mostly due to sentence alignment problems but also due to sloppy translations). The results also clearly indicate that the w-BLEU scores cited in the previous section clearly do not correspond to Human ranking. The morpheme-based reordered model having a lower BLEU score performed better than the baseline system.

6.2 *Error Analysis*

Besides the shortcomings of the evaluation metrics and the corpus itself, there are several real errors emerging during the translation process that can be compensated for in some future work.

1. Errors in parsing of the source-side English sentence can also cause problems in determination of the dependency relations, which will result in erroneous application of the reordering rules. In such cases words that were originally at their correct position will land at the wrong place.

2. Problems of the English PoS sequence: if a word has the wrong tag in the sentence that is to be translated, but it always occurred correctly tagged in the training set, then the system is not able to translate it, even if the word itself is not an unknown word. Likewise, if the translation model contains the same word with several possible PoS tags depending on the context, then if the word in the actual sentence gets the contextually wrong tag, its translation will be wrong (see e.g. *whisper* tagged as a verb (following a determiner!) and thus translated as a verb in Table 7). Tagging errors in the training corpus may result in wrong translation even if the actual parse is correct. Moreover, an incorrect PoS tag usually results in an erroneous syntactic analysis and wrong reordering.

Table 7. The effect of parsing errors

Original sentence:	For 50 years, barely a whisper.
Reordered sentence:	50/[CD] year/[NN] [PL] For/[IN] ,/[,] barely/[RB] a/[DT] **whisper/[VB]** ./[.]
Translated sequence:	50/[NUM_DIGIT] év/[N] [PL] [TER] ,/[PUNCT] alig/[ADV] egy/[DET] **suttog/[V] [S3]** ./[PUNCT]
Morpheme-based:	50 évekig, alig egy **suttog.**
Back-translation:	For 50 year, hardly a **he whispers.**
Baseline:	50 éve, alig egy suttogás.
Back-translation:	50 years ago, hardly a whisper.
Reference:	50 évig a sóhajtásukat sem hallottuk.
Back-translation:	For 50 years, we haven't heard a whisper from them.

3. The quality of the training and test sets has an immediate effect on the measured quality of the translation. The problem is not only that the translation model contains wrong translations learnt from the corpus, but the evaluation metrics compares the results to wrong reference translations. Although this affects translations generated by both the baseline and the morpheme-based system, this might play a role in BLEU score differences not corresponding to how humans rank the translations.
4. Since the smallest units of the translation are morphemes, some of them might be moved to a wrong position. It is often the case in longer sentences that instances of the same functional morpheme belong to more than one different word in the sentence. This causes indeterminacies in the alignment process (because the models imple-

mented in the Giza++ word aligner cannot be forced to assume locally monotonous alignment at the places where we in fact know that alignment should be monotonous) and this usually results in erroneous phrases being extracted from the training corpus. For example if there are two nouns in a sentence, one of them is plural, then the [PL] tag corresponding to this feature might land at another noun.

7 CONCLUSION

In this paper, we described a hybrid phrase-based translation system from English to Hungarian that is an extension of the baseline statistical methods by applying syntax and morphology-based preprocessing steps on the training corpus and morphological postprocessing during translation. The goal was to transform the source-side English sentences to a syntactic structure that is more similar to that of the target-side Hungarian sentences. We concentrated on syntactic structures that have systematically differing realizations in the two languages. We found that readability and accuracy of the translation are improved by the process of reordering the source sentences prior to translation, especially in the cases when the somewhat fragile PoS tagger–parser chain does not lead to wrongly reordered sentences, which has a deteriorating effect on translation quality. Although automatic evaluation assigned the morpheme-based system a significantly and consistently lower score than the baseline system, human evaluation found our systems better than the baseline. We found that several linguistic phenomena can be translated with a much better accuracy than using a traditional SMT system. We also described some problems that are to be solved in the future with the expectation of having an even stronger effect on translation quality.

ACKNOWLEDGEMENT This work was partially supported by TÁMOP – 4.2.1.B – 11/2/KMR-2011-0002.

REFERENCES

1. Lü, Y., Huang, J., Liu, Q.: Improving Statistical Machine Translation performance by training data selection and optimization (2007)
2. Oravecz, C., Dienes, P.: Efficient stochastic part-of-speech tagging for Hungarian. In: LREC, European Language Resources Association (2002)
3. Marneffe, M.C.D., Maccartney, B., Manning, C.D.: Generating typed dependency parses from phrase structure parses. In: In LREC 2006. (2006)

4. Och, F.J., Tillmann, C., Ney, H., Informatik, L.F.: Improved alignment models for Statistical Machine Translation. In: University of Maryland, College Park, MD. (1999) 20–28
5. Collins, M., Koehn, P., Kučerová, I.: Clause restructuring for statistical machine translation. In: Proceedings of the 43rd Annual Meeting of Association for Computational Linguistics. ACL '05, Stroudsburg, PA, USA, Association for Computational Linguistics (2005) 531–540
6. Gojun, A., Fraser, A.: Determining the placement of German verbs in English-to-German SMT. In Daelemans, W., Lapata, M., Màrquez, L., eds.: EACL, The Association for Computer Linguistics (2012) 726–735
7. Yeniterzi, R., Oflazer, K.: Syntax-to-morphology mapping in factored phrase-based statistical machine translation from English to Turkish. In: Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics. ACL '10, Stroudsburg, PA, USA, Association for Computational Linguistics (2010) 454–464
8. Halácsy, P., Kornai, A., Németh, L., Sass, B., Varga, D., Váradi, T., Vonyó, A.: A Hunglish korpusz és szótár [Hunglish corpus and dictionary]. In: III. Magyar Számítógépes Nyelvészeti Konferencia, Szeged, SZTE (2005) 134–142
9. Orosz, G., Novák, A.: PurePos – an open source morphological disambiguator. In: Proceedings of the 9th International Workshop on Natural Language Processing and Cognitive Science., Wroclaw, Poland (2012)
10. Prószéky, G., Kis, B.: A unification-based approach to morpho-syntactic parsing of agglutinative and other (highly) inflectional languages. In: Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics. ACL '99, Stroudsburg, PA, USA, Association for Computational Linguistics (1999) 261–268
11. Novák, A.: What is good Humor like? In: I. Magyar Számítógépes Nyelvészeti Konferencia, Szeged, SZTE (2003) 138–144
12. Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., Bojar, O., Constantin, A., Herbst, E.: Moses: Open Source Toolkit for Statistical Machine Translation. In: Proceedings of the ACL 2007 Demo and Poster Sessions, Prague, Association for Computational Linguistics (2007) 177–180
13. Clifton, A., Sarkar, A.: Combining morpheme-based machine translation with post-processing morpheme prediction. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1. HLT '11, Stroudsburg, PA, USA, Association for Computational Linguistics (2011) 32–42
14. Callison-Burch, C., Fordyce, C., Koehn, P., Monz, C., Schroeder, J.: (meta-) evaluation of machine translation. In: Proceedings of the Second Workshop on Statistical Machine Translation, Prague, Czech Republic, Association for Computational Linguistics (June 2007) 136–158

LÁSZLÓ J. LAKI
MTA-PPKE LANGUAGE TECHNOLOGY RESEARCH GROUP
AND FACULTY OF INFORMATION TECHNOLOGY,
PÁZMÁNY PÉTER CATHOLIC UNIVERSITY,
50/A PRÁTER STREET, 1083 BUDAPEST, HUNGARY
E-MAIL: <LAKI.LASZLO@ITK.PPKE.HU>

ATTILA NOVÁK
MTA-PPKE LANGUAGE TECHNOLOGY RESEARCH GROUP
AND FACULTY OF INFORMATION TECHNOLOGY,
PÁZMÁNY PÉTER CATHOLIC UNIVERSITY,
50/A PRÁTER STREET, 1083 BUDAPEST, HUNGARY
E-MAIL: <NOVAK.ATTILA@ITK.PPKE.HU>

BORBÁLA SIKLÓSI
FACULTY OF INFORMATION TECHNOLOGY,
PÁZMÁNY PÉTER CATHOLIC UNIVERSITY,
50/A PRÁTER STREET, 1083 BUDAPEST, HUNGARY
E-MAIL: <SIKLOSI.BORBALA@ITK.PPKE.HU>

Information Extraction and Biomedical Applications

IJCLA VOL. 4, NO. 2, JUL-DEC 2013, PP. 81–99
RECEIVED 07/12/12 ACCEPTED 04/03/13 FINAL 14/05/13

SnoMedTagger: A Semantic Tagger for Medical Narratives

SAMAN HINA,[1,2] ERIC ATWELL,[1] AND OWEN JOHNSON[1]

[1] *University of Leeds, UK*
[2] *NED University of Engineering & Technology, Pakistan*

ABSTRACT

The identification and classification of semantic information in medical narratives is critical for various research applications such as question-answering systems, statistical analysis, etc. Our contribution is a novel semantic tagger named SnoMedTagger to tag complex semantic information (paraphrases of concepts, abbreviations of concepts, complex multiword concepts) with 16 SNOMED CT semantic categories in medical narratives. SnoMedTagger is developed to support domain users as well as non-domain users working on research questions using medical narratives. Our method includes corpus-based rule-patterns from real world dataset and rule-patterns developed by refinement of SNOMED CT (Systemised NOmenclature of MEDicine-Clinical Terms) clinical vocabulary. These rule-patterns were able to identify semantic information in a range of text and classify them with respective semantic categories derived from SNOMED CT. On unseen gold standard, our rule-pattern-based semantic tagger outperformed SVM-based machine learning system and Ontology-based Bioportal web annotator. The study has shown that it is possible to identify and classify complete semantic information with SNOMED CT semantic categories in medical narratives with high accuracy than achieved by existing approaches.

1 Background

The objective of this research was to develop a generic semantic tagger for identification and classification of semantic information in medical narratives. The presented tagger not only identifies complex multiword concepts, paraphrases of concepts and abbreviation of concepts but also provide a complete tagset extracted from SNOMED CT clinical vocabulary for classification of concepts. Researchers working in medical domain use different names for synonymous semantic categories for their specific research questions. For instance, semantic category 'Test' can also be referred to as 'Procedure' or semantic category 'Treatment' can also be named as 'Medications', which do not follow any standard names used in healthcare data standards. The SNOMED CT tagset used in this framework is customisable and can be used for classification of required semantic categories for various research applications using medical narratives. Because this tagset contains 16 semantic categories derived from international healthcare data standard SNOMED CT, therefore provide consistent information exchange among researchers with globally known semantic categories. SNOMED CT is globally the most comprehensive clinical vocabulary and is specified in several US standards (Stearns et al., 2001).

The classification of medical entities ('X-Ray', 'depression', 'No cough', etc.) with their semantic categories ('Procedures', 'Disorder', 'Findings', etc.) plays an important role in domain specific research. This semantic classification requires domain expertise which is time consuming and expensive; language researchers/non-domain researchers are dependent on domain experts to identify and/or annotate/classify domain specific information. In addition to this, it is also true that output of this approach, i.e., the annotated domain knowledge is restricted for specific research question(s) and therefore, cannot be reused by other researchers.

Many researchers developed biomedical named entity recognition taggers for classification of biomedical texts (Jonquet et al., 2009, Settles, 2005, Seth et al., 2004, Reeve and Han, 2007, Ananiadou et al., 2011). Some used SVM to identify and classify named entities in biomedical text (Zhenfei et al., 2011). Researchers mainly focused on the identification and classification of named entities using journal articles or MEDLINE abstracts but very few work is done on medical narratives with limited classification categories (Meystre et al., 2008).

Thus, there is a need to identify and classify not only named entities but complete semantic information in medical narratives. Medical narratives here refer to discharge summaries, progress notes, etc., written by clinicians whereas biomedical text refer to text in journal articles, MEDLINE abstracts, etc (Meystre et al., 2008). In medical narratives, clinicians express different concepts using semantics ('abbreviations', 'paraphrases', 'complex multi-word', etc.).

Researchers working on domain specific data have to spend considerable amount of resources in designing annotation guidelines and in hiring domain experts to identify and classify the required semantic categories in their dataset such as (Roberts A, 2007, Ohta et al., 2002, Wang, 2007). In automatic approaches, some researchers used linguistic patterns or ontologies to identify limited number of named entities in medical domain (Ogren et al., 2008, Mehdi Embarek and Ferret., 2008, Settles, 2005). Khare et al. (2012) performed contextual and structural analysis for mapping information on forms designed by clinicians with SNOMED CT concepts which is not suitable for unstructured information present in medical narratives.

Existing state-of-the-art systems such as Metamap and Bioportal provide ontologies for identification and classification of concepts in medical domain (Aronson, 2001, Noy et al., 2009a) and it has also been reported that Metamap does not perform well with medical narratives even with the use of extended modules (Meystre et al., 2008). In summary, the existing systems suffer from one or more limitations including failure at complex level of synonymy (Ogren et al., 2008), focus on any specific research question, corpus, limited number of semantic categories using controlled vocabularies/ontologies.

The identification and classification of semantic information from ever increasing number of medical narratives in patient records is critical and challenging for several research applications such as statistical analysis, question-answering systems, negation detection, relationship extraction, etc. In particular, we do not focus on mapping concepts with SNOMED CT controlled vocabulary but use SNOMED CT to classify concepts with semantic categories derived from SNOMED CT. This identification and classification will provide a consistent information exchange to domain users (medical/biomedical researchers) as well as non-domain users (language researchers).

As mentioned earlier, one of the major challenges is to cope with the informal writing structure which can vary from one clinician to another.

These variations in writing styles include the use of abbreviations, complex multi-word concepts, paraphrases of the concepts, etc (with/without use of punctuations). Thus, there is a need for a generic and comprehensive semantic tagger for medical narratives which should be flexible for a range of research questions and enables user to select semantic category according to their requirement. The present work describes the compilation of rule-pattern-based semantic tagger named SnoMedTagger by refinement of the international healthcare data standard SNOMED CT (version 2011) and analysis of real time dataset. Refinement of SNOMED CT was required because of limited writing structure of concepts in vocabulary. The evaluation proved that the SnoMedTagger is able to identify and classify concepts along with SNOMED CT semantic categories in medical narratives covering individual concepts as well as complete concept phrases.

In this paper, first we present the use of SNOMED CT semantic categories used in this research and the development of gold standard corpus for evaluation. Second, we describe the experimental setup of SnoMedTagger, SVM using uneven margins and existing Bioportal web annotator. Lastly, we present the evaluation of all systems against unseen gold standard test dataset and will discuss limitations and future directions.

2 Resources and Gold Standard Corpus

2.1 *Use of SNOMED CT*

In the present study, medical narratives were processed by Bioportal[1] 'Recommender' of ontologies and found the 'SNOMED CT' as best recommendation for medical narratives. SNOMED CT data standard was used for the following reasons; 1) The extraction of all concepts with their semantic categories from 'concept' table to develop a SNOMED CT dictionary application which was used to pre-annotate the corpus for the development of gold standard, 2) The refinement of 'SNOMED CT dictionaries' (explained in Section 3.1) which were used as base vocabulary and used in the development of rule-patterns for

[1] http://bioportal.bioontology.org/recommender

SnoMedTagger (SNOMED CT semantic tagger). Out of 31 top level concept classes and their sub-classes from SNOMED CT (Hina et al., 2010), concepts associated with 16 semantic categories (Attribute, Body Structure, Disorder, Environment, Findings, Observable Entity, Occupation, Person, Physical Object, Procedure, Product or Substance, Qualifier Value, Record Artifact, Regime/Therapy, Situation) were found in medical narratives used in this research. The remaining 15 semantic categories were missed due to following reasons;

- The semantic categories such as 'Physical force', 'Religion', 'Life-style', 'Staging and scales', etc were not found in the corpus used in this research. The concepts associated with these categories refer to special cases which can hardly exist in medical narratives.
- The concepts associated with the semantic categories such as 'Administrative concept', 'Link assertion' (For example; Has problem name, Has problem member etc), 'Namespace concept' (For example; Extension Namespace (1000145)), 'Inactive concept' (consists of outdated concepts, ambiguous concepts, etc), etc were to link and describe the other semantic categories in SNOMED CT data standard.

Particularly, we are not disambiguating the semantic categories in this research because some semantic categories ('Procedure – Regime/Therapy', 'Disorder – Findings') are closely related to each other. For instance, 'Regime/Therapy' is subclass of 'Procedure', 'Disorder' and 'Findings' are subclasses of 'Clinical findings' but according to domain experts may/may not be used as synonym in medical narratives and therefore should be classified separately. Also, it must be noted that semantic type named 'Product or Substance' is the combination of two separate top-level semantic categories, 'Pharmaceutical Product' and 'Substance' which were found synonymous in medical narratives.

2.2 *Development of the gold standard corpus*

The corpus used in this research was categorised into development dataset and test dataset. The development dataset was obtained from the fourth i2b2/VA 2010 challenge which contains discharge summaries and progress notes from different healthcare providers. The test dataset was provided by the Leeds Institute of Health Sciences. It consists of medical narratives written by medical students in a lab session in which a consultation video was shown and the students recorded this consultation in 'System One', an EMR (Electronic Medical Record) system. Recorded narratives were then randomly extracted from the system to create an unseen test dataset. The medical narratives in test dataset were suitable to test the applicability of rule-patterns of semantic tagger as well as to evaluate the performance of the other two systems (SVM-based system, Bioportal web annotator).

The gold standard development dataset and test dataset were annotated following an instruction manual. The instruction manual was designed by authors, considering language issues identified in (Hina et al., 2011). This annotation scheme followed semi-automatic method which is feasible, cheaper and faster compared to manual annotation. This helped both types of users to complete the annotations on time. Two domain users annotated both datasets (development dataset and test dataset) independently following same annotation scheme. The inter-annotator agreement (IAA) was calculated between double annotated datasets as described by (Roberts A, 2007). The inter-annotator agreement for the gold standard development dataset and test dataset was very high and the disagreements were reviewed by a third domain expert. Test dataset was annotated in less time due to less number of concepts and achieved higher IAA than development dataset. Thus, the final gold standard for both datasets was compiled in a consensus set by adding disagreed concepts reviewed by third domain expert. Table 1 shows inter annotator agreement (IAA) and total number of SNOMED CT concepts in the final development and test dataset.

Table 1. Inter annotator agreement and number of annotated SNOMED CT concepts in gold standard development and test dataset

Gold Standard	IAA (%)	Concept annotations in final gold standard
Development dataset	86	5125
Test dataset	95.25	2672

3 Experimental Setup

This section includes the development of SnoMedTagger along with the implementation of the other two systems (SVM based supervised machine learning system, Bioportal web annotator) for evaluation.

3.1 *SnoMedTagger: SNOMED CT Semantic Tagger*

SnoMedTagger is a novel and comprehensive rule-pattern-based semantic tagger for the identification and classification of individual concepts, paraphrases of concepts, abbreviations of concepts and complex multiword concepts along with their SNOMED CT semantic categories in medical narratives. For the development of rule-patterns for semantic tagger, the dictionaries of 16 semantic categories were refined to develop rule-patterns for SnoMedTagger (explained in next section). Although rule-based approach require manual effort, still is effective in absence of large annotated corpus.

Refinement of SNOMED CT concepts for detecting individual concepts and abbreviations

For our purposes, we defined refinement as simplification of multiword concepts, separation of abbreviations from their definitions and removal of unnecessary concepts which are not used by clinicians. The dictionaries of semantic categories derived from SNOMED CT were refined in order to develop generic rule-patterns for SnoMedTagger. In following examples of refinement, all semantic categories are italicised while '→' represents refinement process.

Case 1: Removing unnecessary words and descriptions from SNOMED CT 'Concept' table

In SNOMED CT concept file, several multiword concepts contain descriptive information associated with them. Clinicians do not write this descriptive information in medical narratives and therefore it should be removed for accurate information extraction. Examples of removing descriptions such as 'NOS ', '[SO]', 'NEC', (structure) are as follows.

Example 1:
SNOMED CT concept: Skin NOS – *Body Structure*
Here, NOS = Not otherwise specified
Skin NOS – *Body Structure* → Skin – *Body Structure*

Example 2:
SNOMED CT concept: Vitreous membrane (structure) – *Body Structure*
Vitreous membrane (structure) – *Body Structure* → Vitreous membrane – *Body Structure*

Case 2: Simplification of multiword concepts into individual concepts

Multiword concepts were simplified into individual concepts to produce general rules for SnoMedTagger application following the steps shown below.

Example: SNOMED CT concept:
Entire Skin of Eyelid – *Body Structure*
Step 1: Entire Skin of Eyelid – *Body Structure* →
1) Entire Skin – *Body Structure*
2) Eyelid – *Body Structure*
Step 2: Entire Skin – *Body Structure* →
1) Entire – *Qualifier Value*
2) Skin – *Body Structure*

Case 3: Separation of abbreviations with their descriptions

Several studies reported the extraction of acronyms and abbreviations in biomedical text mainly MEDLINE abstracts using pattern-based approaches and regular expressions (Pustejovsky et al., 2001b, Pustejovsky et al., 2001a, Schwartz and Hearst, 2003). (Nadeau and Turney, 2005) adopted supervised machine learning approach for the identification of acronym-definition pair in biomedical text. (Ao and Takagi, 2005) proved corpus-based algorithm for the identification of abbreviations from MEDLINE abstracts.

In contrast, it was observed that clinicians prefer to write either short form (abbreviation) or long form (definition) in medical narratives. SNOMED CT contains abbreviations along with their definitions in the ontology and also stores this information separately which restrict writing styles in medical narratives.

For this reason, example case described here involves separation of abbreviations from their definitions for each respective dictionary. For instance, SNOMED CT concept: DVT – Deep venous thrombosis or DVT or Deep venous thrombosis can be written in other several possible forms; DVT – (Deep venous thrombosis), DVT (Deep venous thrombosis), (Deep venous thrombosis), DVT, Deep venous thrombosis, (Deep venous thrombosis) DVT, DVT (Deep venous thrombosis), (DVT), DVT: Deep venous thrombosis, Deep venous thrombosis: DVT.

Such concept and similarly other concepts containing abbreviation were simplified as follows:

DVT – Deep venous thrombosis – *Disorder* →
1) DVT – *Disorder*
2) Deep venous thrombosis – *Disorder*

However, there were no examples of abbreviation-definition pair in the development dataset, several pattern-based rules were developed to generalise SnoMedTagger on other datasets (medical narratives). The refinement of SNOMED CT dictionaries is an intermediate stage to apply generic rules for the extraction of semantic information from medical narratives.

System Flow of SnoMedTagger

SnoMedTagger application was developed using GATE - General Architecture for Text Engineering. GATE is an open-source natural language processing software which includes CREOLE: Collection of Reusable Objects for language engineering (Gaizauskas et al., 1996). CREOLE components were used to carry out basic language processing tasks (tokenisation, sentence splitting, part-of-speech (POS) tagging), morphological analysis, and gazetteers/dictionaries. Java Annotation Patterns Engine - JAPE transducers (Cunningham et al., 2000) were used to write rule-patterns for each SNOMED CT semantic category. SnoMedTagger application used 18 CREOLE components and 15 of them were based on JAPE transducers for the development of rules for 15 semantic categories (excluding 'Attribute'), as shown in Fig. 1.

The SnoMedTagger application pipeline first apply basic langauge processing resources (tokensiser, sentence splitter, part-of-speech tagger (Hepple, 2000)) on corpus.

Then, GATE processing resource called flexible gazetteer was used in SnoMedTagger pipeline for the detection of singular as well as plural concepts from refined SNOMED CT dictionaries (explained in earlier section). The flexible gazetteer provides the flexibility to customise the output of refined SNOMED CT dictionaries by morphological analysis. For detection of plural concepts, we used root feature of tokens.

After the identification of both singular and plural concepts with their respective semantic categories, set of rules were added in the SnoMedTagger. Semantic category 'Attribute' does not require rules; therefore rules were developed for the remaining 15 semantic categories.

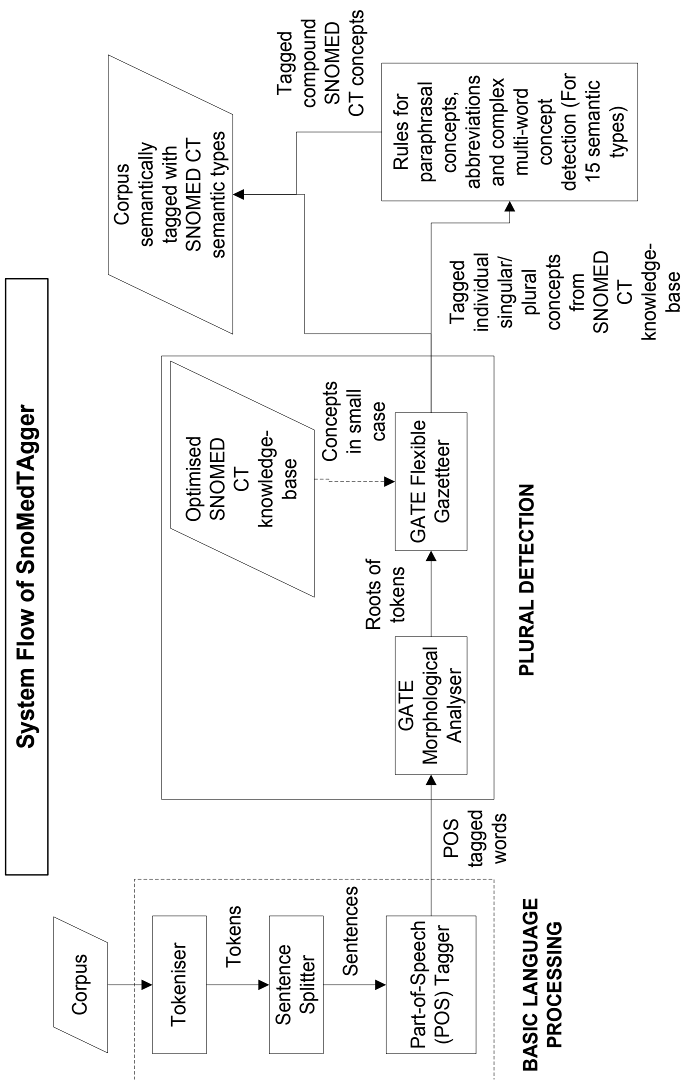

System Flow of SnoMedTAgger
Corpus
Tokeniser
Tokens
Sentence Splitter
Sentences
Part-of-Speech (POS) Tagger
BASIC LANGUAGE PROCESSING
POS tagged words
GATE Morphological Analyser
Roots of tokens
GATE Flexible Gazetteer
Optimised SNOMED CT knowledge-base
Concepts in small case
PLURAL DETECTION
Tagged individual singular/ plural concepts from SNOMED CT knowledge-base
Rules for paraphrasal concepts, abbreviations and complex multi-word concept detection (For 15 semantic types)
Tagged compound SNOMED CT concepts
Corpus semantically tagged with SNOMED CT semantic types

Fig. 1. Application pipeline of SnoMedTagger

This section will explain the development of rule-patterns for the identification and classification of paraphrase concepts, abbreviation of concepts and complex multiword concepts in medical narratives. The derivation of quality rules was from two resources; 1) Analysis of the SNOMED CT data standard. 2) Language of medical narratives written by clinicians.

SNOMED CT data standard contains description logic which is meant to define ontology but has limitation of identifying concepts in medical narratives because of variation in writing styles. Therefore, the rule-patterns were written by analysing real world dataset (development dataset) and rule-patterns analysed during the refinement of SNOMED CT dictionaries. Rules-patterns were written as follows; Rule-pattern --> Rule-action. Example 1 show rule-patterns written by analysing language in SNOMED CT and example 2 contains rule-patterns written by analysing development dataset, where all the semantic categories are italicised. The other notations used in the examples are as follows:

sp= Space Token
IN= Preposition or sub coordinating conjunction
DT= Determiner
|=Or
Lookup.majorType = Bodystructure (dictionary of individual body structures such as 'chest', 'pelvis', 'leg', 'abdomen', etc.)
Lookup.majorType = Procedure (dictionary of individual procedures such as 'X-Ray', 'radiography', 'CT scan', 'biopsy', etc.)
Lookup.majorType = Qualifier_value (dictionary of individual qualifier values such as 'left', 'right', 'upper', 'lower', etc.)

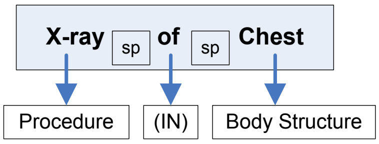
Example 1:
SNOMED CT Concept:
'Radiography of chest' should be marked as *Procedure* and it can be written in several ways:

X-Ray of the chest
Chest X-Ray
Chest x-ray
Radiography of the chest
X-Ray of chest
X-ray of chest
CXR

The individual concepts such as radiography, X-ray, x-ray, X-Ray were marked by dictionaries/gazetteers while for the identification of multiword concepts and paraphrases, following rule-patterns were written using dictionaries and linguistic features analysed in the development corpus.

```
Rule: Procedure
{
(Lookup.majorType=Procedure) (sp) (IN) (sp) (Lookup.majorType=Body structure) |
(Lookup.majorType=Procedure) (sp) (IN) (sp) (DT) (sp) (Lookup.majorType=Body
                                                                  structure) |
(Lookup.majorType=Body structure) (sp) (Lookup.majorType=Procedure)
}: label
-->
:label.Procedure= {Rule=Procedure}
```

For instance, first pattern in this rule can be described as follows:



These rule-patterns are general and will extract other concepts such as; 'GI Prophylaxis', 'pelvic lymphadenectomy', 'abdomen x-ray', 'Prostate biopsy','X-Ray of abdomen' and so on.

Example 2*:*
Below are some corpus-based rule-patterns analysed for the semantic category *Body structure*.

```
Rule: Bodystructure
{
(Lookup.majorType= Bodystructure) (sp) (IN) (sp) (Lookup.majorType= Body structure) |
(Lookup.majorType= Bodystructure) (sp) (IN) (sp) (DT) (sp) (Lookup.majorType= Body
                                                                      structure) |
(Lookup.majorType= Qualifiervalue) (sp) (Lookup.majorType= Bodystructure)
}:label
-->
:label.BodyStructure= {Rule=BodyStructure}
```

These general rule-patterns successfully identified concepts such as 'abdomen of the pelvis', 'Left leg', 'upper quadrant of the belly', 'left eye', 'chest wall', 'second toe on the right foot', 'left ventricular wall thrombus', etc. Similarly, N=316 generic rule-patterns have been written for the 15 semantic categories by analysing all possible combinations of refined SNOMED CT dictionaries and linguistics features, shown in Table 2.

Table 2. Successful combinations of refined dictionaries and linguistic features used in the development of rule-patterns for SnoMedTagger. Shown are the 15 SNOMED CT semantic categories for which rules were developed.

			Body Structure	Disorder	Environment	Findings	Observable Entity	Occupation	Organism	Person	Physical Object	Procedure	Product or	Qualifier Value	Record Artifact	Regime /Therapy	Situation
Successful features used in the development of Rule-Patterns	Token.features	Punctuation	■	■	■	■	■	■	■	■	■	■		■		■	
		IN	■	■	■	■	■	■	■	■	■	■			■	■	■
		DT	■	■	■	■	■	■	■	■	■	■			■	■	■
		TO	■	■	■	■		■			■	■		■	■	■	
		CC	■					■	■		■	■		■			
		JJ		■													
		VBG			■							■					
		VBN			■												
	Refined SNOMED CT semantic categories	Attribute		■		■	■				■						
		Body Structure	■	■		■	■				■	■		■		■	■
		Disorder		■	■					■		■			■	■	■
		Environment			■	■		■		■	■	■		■	■		
		Findings				■						■			■		■
		Observable Entity				■	■										■
		Occupation			■	■		■				■		■	■		
		Organism		■					■								
		Person		■	■					■		■		■	■	■	
		Procedure		■			■			■		■		■	■		■
		Physical Object			■	■		■		■	■	■					
		Product or Substance		■		■	■	■				■	■			■	
		Qualifier Value	■	■	■	■				■		■					■
		Record Artifact												■	■		
		Regime /Therapy														■	
		Situation															■
	LEGEND: Highlighted boxes indicate used features																

3.2 *Using Supervised Machine Learning for Semantic annotation*

To evaluate the performance of our rule-based approach against machine learning, we used Java version of Support Vector Machines (SVMs) package LibSVM with uneven margins (Li and Shawe-Taylor, 2003). SVM is known for classification in language processing tasks and learns all features with high generalisation using kernel function. We used linear kernel with the extension of multiple classification ('one Vs others'). The general feature set used in the development of patterns was also used to train the classifier on development dataset (training set). The training was completed using following feature set.

1. Refined SNOMED CT dictionaries (for chunking individual concepts).
2. Part-of-speech categories of three words before and three words after dictionary terms.
3. Three Words before and three words after the roots of the token
4. The type/kind of tokens for learning punctuations 4 words before and 4 words after the term. These ranges were provided in order to learn long and complex multi-word concepts from the development corpus. The results were then compared against gold standard test dataset, described in section 4. Results showed that it is difficult to achieve high recall using general features for all 16 semantic categories.

3.3 *Bioportal Web Annotator*

Bioportal is a web portal which provides a selection of over 300 ontologies from biological and medical domain (Noy et al., 2009b). In this research, bioportal 'recommender[2]' was used for the recommendation of SNOMED CT ontology for medical narratives and then bioportal web annotator was used to annotate test dataset with selection of 16 SNOMED CT categories used in this research. Bioportal provide python client code which was used to annotate the test dataset using SNOMED CT ontology.[3] The annotations were then compared against human annotated gold standard presented in results section.

[2] http://bioportal.bioontology.org/recommender

[3] http://www.bioontology.org/wiki/index.php/Annotator_Web_service

4 Evaluation

The SnoMedTagger was developed using development dataset that contained concepts associated with 16 semantic categories derived from SNOMED CT; however the 'Organism' semantic category was missing in the gold standard test dataset. To evaluate all the three systems against unseen gold standard test dataset that contained 15 semantic categories, standard metrics (recall, precision, f-measure) were used. We focused on improvement of recall and f-measure of the SnoMedTagger to prove reliability of the rule-patterns. SnoMedTagger overall achieved 82% recall, 71% precision and 76% of f-measure while SVM based system overall achieved 49% recall, 81% precision, 61% f-measure and Bioportal system achieved 52% recall, 40% precision, 45% f-measure. The f-measure of rule-pattern-based SnoMedTagger outperformed the application using SVM with uneven margins (SVM-UM) and the ontology-based Bioportal web annotator. The application using SVM with uneven margins has achieved high precision but achieved very low recall because of granularity levels (identification of concept phrases).

On the other hand, ontology-based Bioportal web annotator predictably achieved low scores in all three systems because of inappropriateness of controlled language of ontology. This proved that the language used in controlled vocabularies is insufficient to identify and classify semantic information in medical narratives. Although, SNOMED CT clinical vocabulary cannot directly incorporated with medical narratives written by clinicians, still served as a useful resource to recognise the gap between controlled vocabularies and medical narratives. On the other hand, it is difficult to achieve general applicability using machine learning approach because it can only perform better in case of similar data (training and test). The overall recall, precision and f-measure for three systems are shown in Fig. 2.

5 Conclusions and Future Work

This paper presented a rule-pattern-based semantic tagger (SnoMedTagger) for the identification and classification of all possible semantic information in medical narratives. SnoMedTagger will facilitate researchers to extract semantic information from medical narratives with the categorisation of SNOMED CT standard semantic categories. The corpus-based rule-patterns and rule-patterns analysed by refining SNOMED CT ensure that the coverage of SnoMedTagger is not only limited to medical narratives but the framework may also be helpful for researchers to analyse the limitation of controlled vocabularies (UMLS, SNOMED CT, ICD-10, etc.) on real world datasets.

We presented the results of our system on unseen test data to prove the general applicability of rule-based SnoMedTagger and also compared the output of two systems (SVM-based system, bioportal web annotator) on the same test dataset. Reasonable accuracy was achieved on unseen test dataset but we still believe in further evaluation of SnoMedTagger on more than one dataset.

Moreover, to improve the accuracy of SnoMedTagger framework, future directions also include the investigation of rules on different test cases from real world datasets and then validation of extracted concepts by getting feedback from different domain experts. We expect to contribute our semantic tagger as open source tool for research purposes.

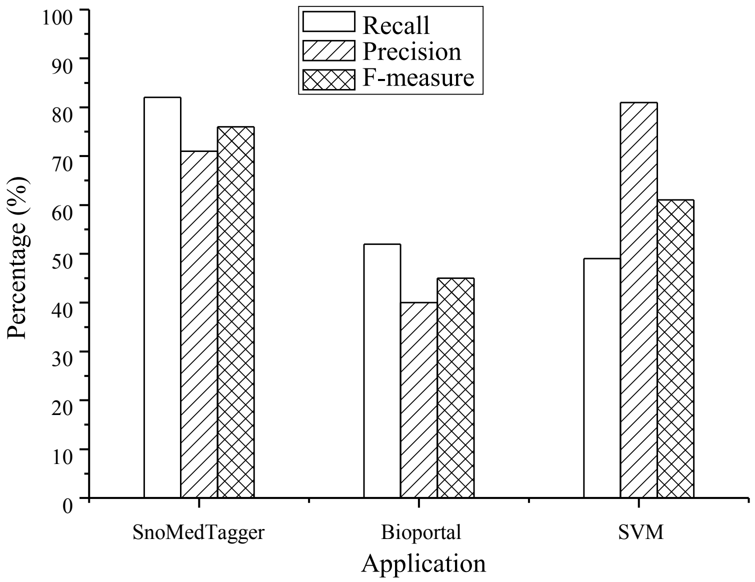


Fig. 2. Evaluation of SnoMedTagger, SVM-UM and Bioportal application against gold standard test dataset

6 References

1. Ananiadou, S., Sullivan, D., Black, W., Levow, G.-A., Gillespie, J. J., Mao, C., Pyysalo, S., Kolluru, B., Tsujii, J. & Sobral, B. 2011. Named Entity Recognition for Bacterial Type IV Secretion Systems. PLoS ONE, 6, e14780.
2. Ao, H. & Takagi, T. 2005. Alice: An Algorithm to Extract Abbreviations from MEDLINE. J Am Med Inform Assoc, 12, 576 - 586.
3. Aronson, A. 2001. Effective mapping of biomedical text to the UMLS Metathesaurus: the MetaMap program. Proceedings AMIA Symposium, 17 - 21.
4. Cunningham, H., Mayard, D. & Tablan, V. 2000. JAPE: a JAVA Annotation Patterns Engine Second Edition ed. Sheffield: University of Sheffield.
5. Gaizauskas, R., Cunningham, H., Wilks, Y., Rodgers, P. & HumphreyS, K. GATE: an environment to support research and development in natural language engineering. Tools with Artificial Intelligence, 1996., Proceedings of Eighth IEEE International Conference 16-19 Nov. 1996 1996. 58-66.
6. Hepple, M. 2000. Independence and Commitment: Assumptions for Rapid Training and Execution of Rule-based Part-of-Speech Taggers. in Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics (ACL-2000).
7. Hina, S., Atwell, E. & Johnson, O. 2010. Semantic Tagging of Medical Narratives with Top Level Concepts from SNOMED CT Healthcare Data Standard. International Journal of Intelligent Computing Research (IJICR), 1, 118-123.
8. Hina, S., Atwell, E. & Johnson, O. Enriching the corpus of Natural Language Medical narratives with healthcare data standard SNOMED CT. Corpus Linguistics, 2011 Birmingham, United Kindom.
9. Jonquet, C., Shah, N. & Musen, M. 2009. The Open Biomedical Annotator. AMIA Summit on Translational Bioinformatics. San Francisco.
10. Khare, R., An, Y., Li, J., Song, I.-Y. & Hu, X. 2012. Exploiting semantic structure for mapping user-specified form terms to SNOMED CT concepts. Proceedings of the 2nd ACM SIGHIT International Health Informatics Symposium. Miami, Florida, USA: ACM.
11. Li, Y. & Shawe-Taylor, J. The SVM with uneven margins and Chinese document categorization. The 17th pacific Asia Conference on Language , Information and Computation (PACLIC17), 2003 Singapore. 216–227.

12. Mehdi Embarek & Ferret., O. Learning Patterns for Building Resources about Semantic Relations in the Medical Domain. In Proceedings of LREC'2008. , 2008.
13. Meystre, S. M., Savova, G. K., Kipper-Schuler, K. C. & Hurdle, J. F. 2008. Extracting information from textual documents in the electronic health record: a review of recent research.
14. Nadeau, D. & Turney, P. 2005. A Supervised Learning Approach to Acronym Identification. In Proceedings of Canadian Conference on AI'2005.
15. Noy, N., Shah, N., Whetzel, P., Dai, B., Dorf, M., Griffith, N., Jonquet, C., Rubin, D., Smith, B., Storey, M., Chute, C. & Musen, M. 2009a. Bioportal: Ontologies and Integrated Data Resources at the Click of a Mouse. Nucleic Acids Res.
16. Noy, N. F., Shah, N. H., Whetzel, P. L., Dai, B., Dorf, M., Griffith, N., Jonquet, C., Rubin, D. L., Storey, M.-A., Chute, C. G. & Musen, M. A. 2009b. BioPortal: ontologies and integrated data resources at the click of a mouse. Nucleic Acids Research, 37**,** W170-W173.
17. Ogren, P., Savova, G. & Chute, C. Constructing Evaluation Corpora for Automated Clinical Named Entity Recognition. LREC, 2008.
18. Ohta, T., Tateisi, Y. & Kim, J.-D. 2002. The GENIA corpus: an annotated research abstract corpus in molecular biology domain. Proceedings of the second international conference on Human Language Technology Research. San Diego, California: Morgan Kaufmann Publishers Inc.
19. Pustejovsky, J., Castaño, J., Cochran, B., Kotecki, M. & Morrell, M. (eds.) 2001a. Automatic Extraction of Acronym-meaning Pairs from MEDLINE Databases.: IOS Press.
20. Pustejovsky, J., Castano, J., Cochran, B., Kotecki, M., Morrell, M. & Rumshisky, A. 2001b. Extraction and disambiguation of acronym-meaning pairs in medline. Medinfo, 10**,** 371-375.
21. Reeve, L. & Han, H. 2007. CONANN: An Online Biomedical Concept Annotator. Lecture Notes in Computer Science, 4544**,** 264.
22. Roberts A, G. R., Hepple M, Davis N, Demetriou G, Guo Y, Kola J, Roberts I, Setzer A, Trapuria A, Wheeldin B. 2007. The CLEF corpus: semantic annotation of clinical text. AMIA Annu Symp Proc**,** 625-629.
23. Schwartz, A. & Hearst, M. 2003. A Simple Algorithm for Identifying Abbreviation Definitions in Biomedical Text. Proceedings of the 8th Pacific Sym-posium on Biocomputing: 03-07 January 2003; Lihue, Hawaii**,** 451–462.

24. Seth, K., Bies, A., Liberman, M., Mandel, M., Mcdonald, R., Palmer, M. & Schein, A. Integrated annotation for biomedical information extraction. Proceedings of the BioLINK 2004, 2004.
25. Settles, B. 2005. ABNER: an open source tool for automatically tagging genes, proteins and other entity names in text. Bioinformatics, 21**,** 3191 - 2.
26. Stearns, M. Q., Price, C., Spackman, K. A. & Wang, A. Y. 2001. SNOMED clinical terms: overview of the development process and project status. Proc AMIA Symp**,** 662–666.
27. Wang, X. 2007. Rule-Based Protein Term Identification with Help from Automatic Species Tagging. Proceedings of the 8th International Conference on Computational Linguistics and Intelligent Text Processing, CICLing 2007. Mexico City, Mexico: Springer-Verlag.
28. Zhenfei, J., Jian, W. & Fei, Z. Named Entity Recognition from Biomedical Text Using SVM. 5th International Conference on Bioinformatics and Biomedical Engineering, (iCBBE) 2011., 10–12 May 2011. 1-4.

SAMAN HINA
SCHOOL OF COMPUTING,
UNIVERSITY OF LEEDS,
UK
AND DEPARTMENT OF CS&IT,
NED UNIVERSITY OF ENGINEERING & TECHNOLOGY,
PAKISTAN
E-MAIL: <SCSH@LEEDS.AC.UK, SAMAN.HINA@GMAIL.COM>

ERIC ATWELL
SCHOOL OF COMPUTING,
UNIVERSITY OF LEEDS,
UK
E-MAIL: <E.S.ATWELL@LEEDS.AC.UK >

OWEN JOHNSON
SCHOOL OF COMPUTING,
UNIVERSITY OF LEEDS,
UK
E-MAIL: <O.A.JOHNSON@LEEDS.AC.UK>

IJCLA VOL. 4, NO. 2, JUL-DEC 2013, PP. 101–120
RECEIVED 07/12/12 ACCEPTED 04/03/13 FINAL 14/03/13

Towards Event-based Discourse Analysis of Biomedical Text

RAHEEL NAWAZ, PAUL THOMPSON, AND SOPHIA ANANIADOU

University of Manchester, UK

ABSTRACT

Annotating biomedical text with discourse-level information is a well-studied topic. Several research efforts have annotated textual zones (e.g., sentences or clauses) with information about rhetorical status, whilst other efforts have linked and classified sets of text spans according to the type of discourse relation holding between them. A relatively new approach has involved annotating meta-knowledge (i.e., rhetorical intent and other types of information concerning interpretation) at the level of bio-events, which are structured representations of pieces of biomedical knowledge. In this paper, we report on the examination and comparison of transitions and patterns of event meta-knowledge values that occur in both abstracts and full papers. Our analysis highlights a number of specific characteristics of event-level discourse patterns, as well as several noticeable differences between the types of patterns that occur in abstracts and full papers.

KEYWORDS: *meta-knowledge, event, bio-event, discourse analysis.*

1 Introduction

The identification of information about the structure of scientific texts has been studied from several perspectives. One line of previous research has been to classify textual zones (e.g., sentences or clauses) according to their function in the discourse, such as background knowledge, hypotheses, experimental observations, conclusions, etc. The automatic identification of such information can help in tasks such as isolating new knowledge claims [1]. Within the biomedical domain, this information can in turn be useful for tasks such as maintaining models of biomedical processes [2] or the curation of biomedical databases [3].

Several annotation schemes, e.g., [4-6] have been developed to classify textual zones according to their rhetorical status or general information content. Such zones are usually not understood in isolation, but rather in relation to others [7]. Therefore, for certain tasks, such as automatic summarisation, it is important to gain a fuller understanding of how information conveyed in the text is arranged to form a coherent discourse. Work in this area has involved defining a model that describes the structure of the introductions to scientific articles [8] and examining patterns of argumentative zones that occur in scientific abstracts [9].

A further approach to discourse analysis has been to identify and characterise links between sentences and clauses. Several efforts to produce annotated corpora or automated systems have been based around the Penn TreeBank corpus of open domain news articles [10]. This corpus was enriched by [11] with discourse trees, based on Rhetorical Structure Theory (RST) [12], A system was created by [7] to predict certain classes of discourse relations automatically. The Penn Discourse TreeBank (PDTB) [13] added discourse relations to the Penn TreeBank, both implicit and explicit, that hold between pairs of text spans. The Biomedical Discourse Relation Bank (BioDRB) [14] annotates the same types of relations in biomedical research articles.

All of the studies above considered sentences or clauses as the units of annotation. In contrast, the present work is concerned with discourse information at the level of *events*, which are structured representations of pieces of knowledge. In particular, we focus on bio-events, which encode biological reactions or processes. The automatic identification of events can facilitate sophisticated semantic searching, allowing researchers to perform structured searches over events extracted from a large body of text [15].

The utility of events has resulted in the appearance of a number of event-annotated corpora in recent years, e.g., [16-18]. The shared tasks on event extraction at BioNLP workshops, e.g., [19] have helped to stimulate further research into event extraction. Since there are normally multiple events in a sentence, the identification of discourse information at the event level can allow for a more detailed analysis of discourse elements than is possible when considering larger units of text.

Previous work on annotating discourse at the level of events has involved defining a customised annotation scheme [20] encoding various aspects of knowledge that can be relevant to discourse. This *meta-knowledge* scheme has been used to enrich the GENIA event corpus of 1,000 biomedical abstracts (36,858 events) [16] to create the GENIA-MK corpus [21], and a corpus of 4 full papers pre-annotated with 1,710 GENIA events to create the FP-MK corpus [22].

The meta-knowledge annotation scheme is somewhat comparable to the sentence-based classification schemes introduced above, in that it includes encoding of specific rhetorical functions, e.g., fact, observation, analysis (referred to as *Knowledge Type* (KT)). However, further types of relevant to discourse analysis. e.g., certainty level (*CL)*, are also annotated for each event. Automatic recognition of different types of meta-knowledge for events has been demonstrated to be highly feasible [23, 24].

The annotation of information about discourse function at the level of events has been shown to be complementary to sentence-based classification schemes [25], meaning that event-based discourse analysis could help to enrich previous efforts to annotate and recognise discourse information using coarser-grained textual units.

In this paper, we describe our preliminary work on analysing the discourse structure of biomedical abstracts and full papers at the level of events. To our knowledge, this is a novel approach to event-level discourse analysis. Specifically, we look at patterns of transitions between events, in terms of *KT* and *CL*, based on the event-level meta-knowledge annotations that are already present in the GENIA-MK and FP-MK corpora. At the sentence/clause level, it has been found previously that it is not possible to apply a fixed model of discourse structure consistently to all scientific texts [9], and hence we also do not attempt this at the event level. Rather, we examine patterns of *KT* and *CL* values assigned to sequences of events of various lengths.

The remainder of this paper is structured as follows. In section 2, we provide further details about events and the meta-knowledge annotation scheme. In section 3, we look at the different types of transitions, both between pairs of adjacent events and for longer paths of events that occur in the abstracts of GENIA-MK corpus. In section 4, we examine the pairwise transitions in the full papers of the FP-MK corpus, while section 5 provides some concluding remarks and directions for future work.

TRIGGER:	***augmented***
TYPE:	**positive_regulation**
THEME:	***c-jun mRNA*** : RNA_molecule
CAUSE:	***LTB4*** : organic_molecule

Fig. 1. Typical representation of the bio-event contained in sentence S1

2 Bio-events and their Enrichment with Meta-knowledge

In this section, we provide a brief introduction to bio-events, and describe the meta-knowledge annotation scheme that has been designed to enrich them with additional information about their interpretation, including discourse-level information.

2.1 *Bio-events*

In its most general form, a **textual event** can be described as an action, relation, process or state expressed in the text [26]. More specifically, it is a structured semantic representation of a piece of information contained in the text. Events are usually anchored to text fragments that are central to the description of the event, e.g., *event-trigger*, *event-participants* and *event-location*, etc. A number of corpora of general language with event-like annotations have been produced, e.g., [27, 28].

A **bio-event** is a specialised textual event, constituting a dynamic bio-relation involving one or more participants [16]. These participants can be bio-entities or (other) bio-events, and are each assigned a semantic role like *theme* and *cause*. Bio-events and bio-entities are also typically assigned semantic types/classes from particular taxonomies/ontologies. Consider the sentence S1: "*We conclude that LTB4 may augment c-jun mRNA*". This sentence contains a single bio-event of type *positive_regulation*, which is anchored to the verb *augmented*. Figure 1 shows a typical structured representation of this bio-event, with two participants: *c-jun mRNA* and *LTB4*, which have been assigned semantic types and roles within the event.

2.2 *Meta-Knowledge*

Whilst Figure 1 shows the typical information that would be extracted from sentence S1 by an event extraction system, there is other infor-

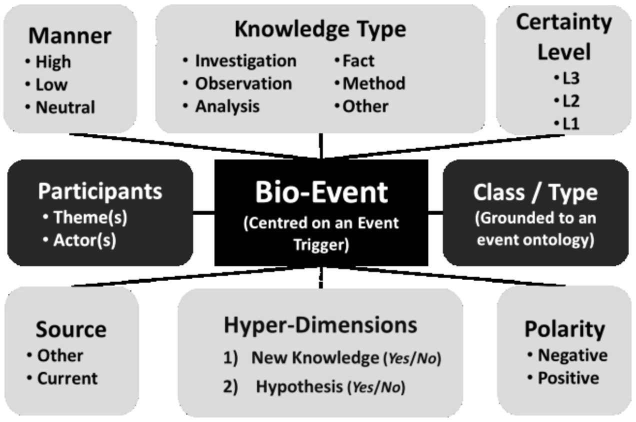


Fig. 2. Meta-knowledge annotation scheme

mation present in S1 that must be extracted if the event is to be interpreted correctly. For example, in terms of *KT*, the event does not represent a definite fact, but rather an analytical conclusion drawn by the authors. Similarly, the presence of the word *may* shows that the conclusion drawn is a tentative one, i.e., the *CL* of the analysis encoded by the event is low. The meta-knowledge annotation scheme (Figure 2) is able to capture this information about the event. The scheme consists of 5 different meta-knowledge dimensions, which encode not only discourse-relevant information, but also other common types of information that are necessary for the correct interpretation of a bio-event.

Due to the complexity of analysing the transitions between the values of all 5 meta-knowledge dimensions, and since not all of the dimensions are directly related to discourse structure, we consider only the two dimensions of the scheme that are most relevant in this respect, i.e. *KT* and *CL.* These are defined as follows:

Knowledge Type (KT)

This dimension captures the general information content of the event. Each event is classified into one of the following six categories:

- **Investigation**: Enquiries or investigations.
- **Observation**: Direct experimental observations
- **Analysis**: Inferences, interpretations, speculations or other types of analysis.
- **Fact:** General facts and well established knowledge.

- **Method:** Events that describe experimental methods.
- **Other:** Default category, assigned to events that either do not fit into one of the above categories or do not express complete information.

Certainty Level (CL)

This dimension is only applicable to events whose KT corresponds to *Analysis*. It encodes confidence in the truth of the event. Possible values are as follows:

- **L3**: No expression of uncertainty or speculation (default category).
- **L2**: High confidence or slight speculation.
- **L1**: Low confidence or considerable speculation.

3 Analysis of Meta-Knowledge Transitions in Abstracts

In this section, we present a brief analysis of the meta-knowledge transitions observed in the GENIA-MK corpus. We begin with patterns of individual, pair-wise transitions and then move on to look at longer transition paths.

3.1 *Knowledge Type (KT)*

Pair-wise Transitions

Figure 3 provides a summary of the pair-wise transitions **from** and **to** adjacent events in the GENIA-MK corpus, according to *KT* categories. The black lines represent the transitions **from** the category in the centre of the diagram), while the grey lines indicate the transitions **to** that category. Similarly, the dark grey boxes show the relative frequencies of each type of transition **from** the category, while the light grey boxes show the relative frequencies of each type of transition **to** the category. The dotted lines boxes surrounded by dotted lines represent reflexive transitions, i.e., cases where the *KT* category of the adjacent event is the same as the event in focus. Transitions between all adjacent pairs of events are taken into account, i.e., not only those occurring within the boundaries of a sentence.

Observation: This is a highly reflexive category, with 80% of transitions from *Observation* leading to another *Observation*; similarly 83% of transitions to an *Observation* originate from another *Observation*. In

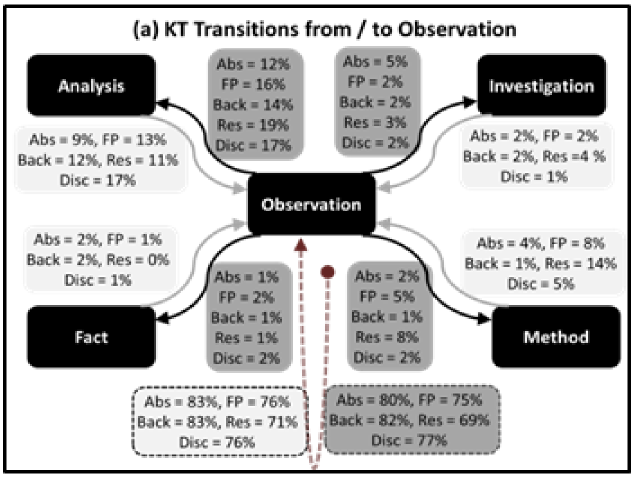


Fig. 3. Transitions from/to *KT* categories for Abstracts (Abs), Full Papers (FP), and the different sections within full papers, i.e., Background (Back), Results (Res), and Discussion (Disc). Continued to the next page.

terms of non-reflexive transitions, 12% of transitions originating from *Observation* lead to *Analysis*, because observations are often used as premises for analytical and hypothetical conclusions. Conversely, most non-reflexive transitions leading to *Observation* start from *Analysis*. This is probably due to the linked nature of arguments presented in an abstract, i.e., the conclusion of an argument can be used as the premise of the next argument. A small but noticeable proportion (5%) of transitions starting from *Observation* lead to *Investigation*. However, in most cases, these observations are attributed to previous studies (as determined by the *Source* dimension of the annotation scheme). That is, a previous observation has been used as a premise for a new investigation.

Analysis: This is also a highly reflexive category, with 70% of the transitions from *Analysis* leading to another *Analysis* and 62% of transitions to *Analysis* originating from *Analysis*. In terms of non-reflexive transitions, 18% of transitions from *Analysis* lead to *Observation* (possible reasons have been discussed above). Similarly, a significant proportion (23%) of transitions that lead to *Analysis* start from *Observation*. Transitions from *Analysis to Fact* are very infrequent (1%). Conversely, 9% of all transitions leading to *Analysis* originate from *Fact*. This is because the state-of-the-art knowledge is sometimes analysed in

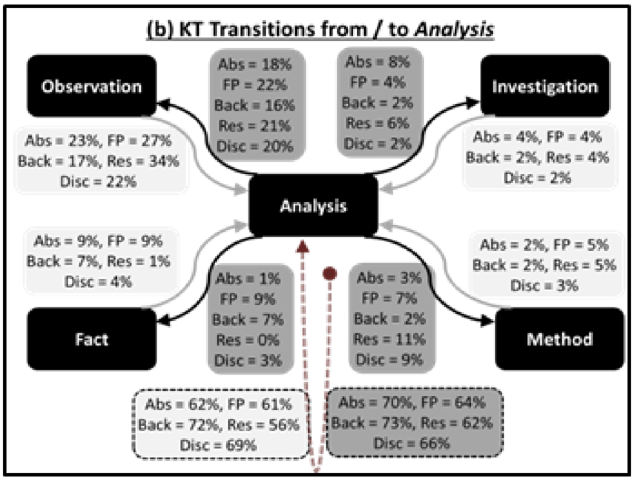

(b) KT Transitions from / to Analysis
Observation
Abs = 18%, FP = 22%, Back = 16%, Res = 21%, Disc = 20%
Abs = 8%, FP = 4%, Back = 2%, Res = 6%, Disc = 2%
Investigation
Abs = 23%, FP = 27%, Back = 17%, Res = 34%, Disc = 22%
Abs = 4%, FP = 4%, Back = 2%, Res = 4%, Disc = 2%
Analysis
Abs = 9%, FP = 9%, Back = 7%, Res = 1%, Disc = 4%
Abs = 2%, FP = 5%, Back = 2%, Res = 5%, Disc = 3%
Abs = 1%, FP = 9%, Back = 7%, Res = 0%, Disc = 3%
Abs = 3%, FP = 7%, Back = 2%, Res = 11%, Disc = 9%
Fact
Method
Abs = 62%, FP = 61%, Back = 72%, Res = 56%, Disc = 69%
Abs = 70%, FP = 64%, Back = 73%, Res = 62%, Disc = 66%

Fig. 3, continued. Continued to the next page.

order to situate or justify the study that is reported in a paper. Further evidence for this pattern is that a similar proportion (8%) of transitions starting from *Analysis* lead to *Investigation*. i.e., in cases where background knowledge is stated and analysed, it is usual that the analysed information is used as a basis for introducing the focussed investigation of the current study.

Investigation: This is a less reflexive category, with only 50% of transitions from *Investigation* leading to other *Investigation*s, and 62% transitions to *Investigation* events originating from other *Investigation*s. This is because the main investigation is usually discussed only at the beginning of the abstract, followed by observations and analyses. This argument is further supported the significant number of transitions from *Investigation* that lead to *Observation* (26%) or *Analysis* (15%).

Fact: This is also a less reflexive category: 63% of all transitions from *Fact* lead to other *Fact*s, and vice versa. *Fact*s are often followed by *Analysis* (19%), as described in the *Analysis* section above. In some cases, *Fact*s serve as direct premises for *Investigation* (10%). Infrequently, *Fact*s are directly followed by *Observation*s (6%).

Method: Only 33% of transitions from/to *Method* are reflexive. In abstracts, authors tend to mention the methods used in their work only briefly (if at all). Since it is natural for authors to move from the de-

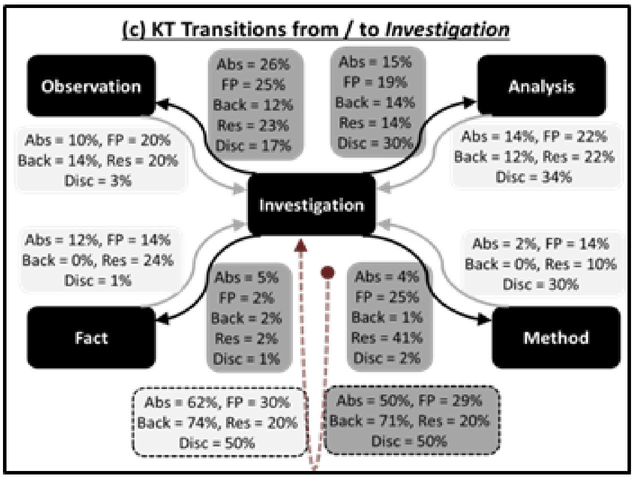


Fig. 3, continued. Continued to the next page.

scription of methods to subsequent experimental results, this explains why the highest proportion of transitions from *Method* events (44%) lead to *Observation* events. However, since the reporting of experimental outcomes or conclusions is of vital importance in abstracts, observations will sometimes be omitted, and authors move straight from describing methods to analysing their findings. This goes towards explaining why 15% of *Method*s are directly followed by *Analysis*. Most of the non-reflexive transitions that lead to *Method* originate from *Observation* (36%). This is because authors frequently present findings from previous studies to set the scene for introducing their own experimental methods. A significant percentage of transitions to *Method* are from *Analysis* (16%). In some cases, an analysis of previous findings is necessary to correctly justify the author's own methods. In other cases, authors complete their discussion of one set of experiments and then move on to introducing a further set of methods.

Abstract Level Patterns

The results of analysing the *KT* values of the first and last event in each abstract are summarised in Table 1. Mostly, authors begin by stating known *Fact*s as a scene-setting device for introducing their own work. The use of *KT* categories other than *Fact* at the start of abstracts is considerably less frequent, with *Analysis* and *Observation* as the next most common categories. Analysis of the *Source* dimension of these event

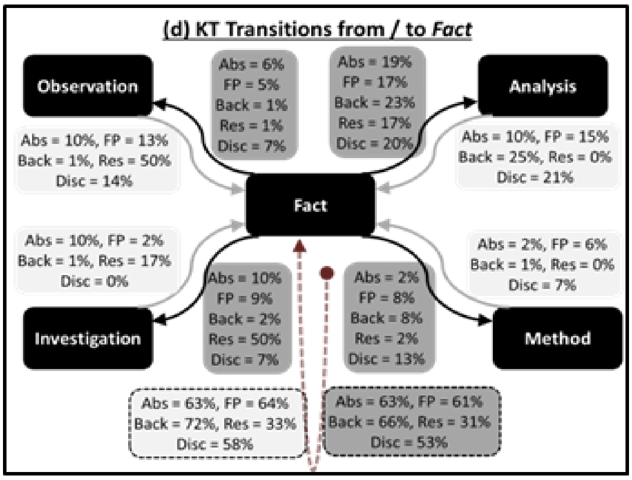


Fig. 3, continued. Continued to the next page.

types reveals that they often pertain to previous studies, indicating that a discussion of previous findings is also a common way to start.

Sometimes, scene-setting steps are omitted altogether, and the abstract launches directly into an explanation of the investigation to be undertaken. In rare cases, even the subject of investigation is missing, and the abstract starts by explaining the experimental setup and methodology. In the vast majority of cases, authors end their abstracts with an *Analysis*, presenting a summary or interpretation of their most important findings. However, there is a significant proportion of cases (15%) in which the abstract ends with an *Observation*. This can happen the when a significant experimental observation has occurred during the current study. Very occasionally, the abstracts end by presenting an investigative topic or method identified for further exploration.

Table 1. Relative frequencies of abstracts starting and ending with each *KT* category

KT Category	Abstracts Starting With	Abstracts Ending With
Observation	10%	15%
Analysis	23%	78%
Investigation	9%	4%
Fact	54%	1%
Method	4%	2%

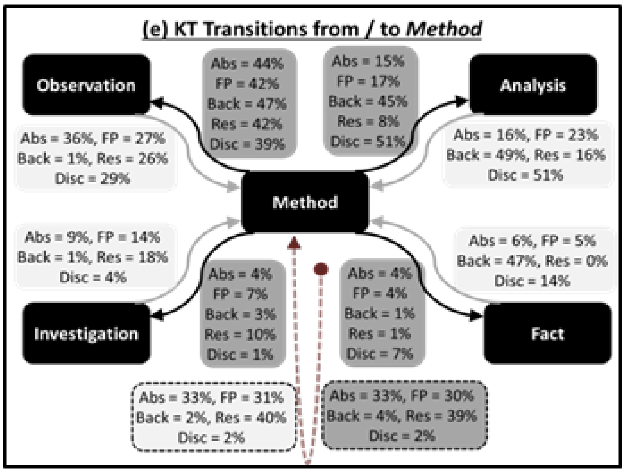


Fig. 3, continued.

Table 2 shows the most frequent extended transition patterns of KT values. Almost a quarter of all abstracts start with known facts, followed by analyses of previous work or a description of the investigation to be carried out in the current study; this is in turn followed by a description of experimental observations, and the abstract ends with an analysis of these observations. Interestingly, over 8% of the abstracts exhibit a simplified variant of this pattern, where the second transition to *Analysis* or *Investigation* is omitted and a direct link is made between the previously known facts and the (new) observations made by the authors. A possible explanation of this could be the need for brevity resulting from the fact that abstract size constraints vary between biomedical journals.

Table 2. Key transition patterns for *KT* values in abstracts and their frequencies

Transition Pattern	% in Abstracts
Fact → Analysis → Observation → ... → Analysis	14%
Fact → Investigation → Observation → ... → Analysis	10%
Fact → Observation → ... → Analysis	8%
Analysis → Observation → ... → Analysis	7%
Analysis → Fact → Observation → ... → Analysis	6%
Analysis → Investigation → Observation → ... → Analysis	4%

A significant number of abstracts follow a slightly different *KT* transition pattern. They start with an analysis of previous studies, followed by observations from the current study, and end with an analysis of findings. Variants of this pattern, which include a transition to a *Fact*, to help to contextualise the analyses of previous studies, or present an *Investigation* between the first *Analysis* and *Observation* events, are also found in 10% of abstracts.

The above patterns suggest that while most biomedical abstracts loosely follow the *Creating A Research Space (CARS)* model proposed by Swales [29], a significant proportion of abstracts skip the first step of "establishing a territory", and assume that the reader is already familiar with the context. This could be due to partly to the specialised nature of many biomedical journals.

3.2 Certainty Level (CL)

Pair-wise Transitions

Figure 4 summarises the pair-wise transitions **from** and **to** adjacent events in the GENIA-MK corpus, according to the *CL* category assigned to them.

<u>L3:</u> This is a highly reflexive category, partly due to its high frequency of occurrence (92% of events in the GENIA-MK corpus). In terms of non-reflexive transitions, 6% of transitions from *L3* lead to *L2*, and only 1% to *L1*. As explained earlier, most abstracts start with a brief mention of previous knowledge (observations, analyses or facts), followed by a summary of investigations and the resulting observations, and conclude with analyses of experimental findings, which are often hedged.

<u>L2:</u> This is the least reflexive category, partly due to the fairly small number of *L2* events in the corpus as a whole. Also, since authors do not want to throw too much doubt on their findings, they avoid long chains of speculated events. This explain why significant proportion (40%) of transitions from *L2* lead back to *L3*. Interestingly, 6% of transitions from *L2* lead to *L1*. These are mostly the cases where slightly hedged analyses are followed by bolder (highly speculative) extensions and corollaries.

<u>L1:</u> For similar reasons as *L2*, this is also a less reflexive category. Although a significant proportion of transitions from *L1* events lead to *L3* (34%) and *L2* (6%) events, the volumes of *L1* events are so small

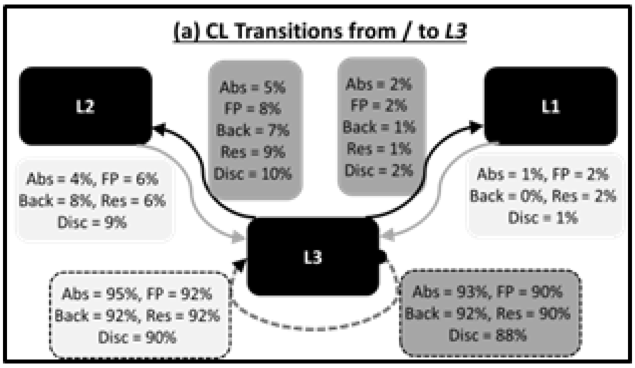


Fig. 4. Transitions from / to *CL* categories for Abstracts (Abs), Full Papers (FP), and the sections within full papers, i.e., Background (Back), Results (Res), and Discussion (Disc). Continued to the next page.

(less than 1% of all events) that they only account for around 1% of all transitions to *L3* and *L2*.

Abstract Level Patterns

The *CL* values of the first and last event in each abstract in the GENIA-MK corpus are summarised in Table 3. Almost all abstracts start with known facts, previous observations, analyses, or investigations, i.e., events expressed with absolute certainty of occurrence (*L3*). Although most abstracts end with analyses, authors will usually aim to have maximum impact at the end of their abstract, so as to encourage reading of the full text.

This means that where possible, hedging will either be absent, or only subtly expressed. A smaller, but still important percentage of terminal events are marked as highly speculative, sine impact can also be achieved by presenting analyses that are both highly speculative and highly innovative or controversial.

Table 3. Relative frequencies of abstracts starting and ending with different *CL* categories

CL Category	Abstracts Starting With	Abstracts Ending With
L1	0%	19%
L2	1%	36%
L3	99%	45%

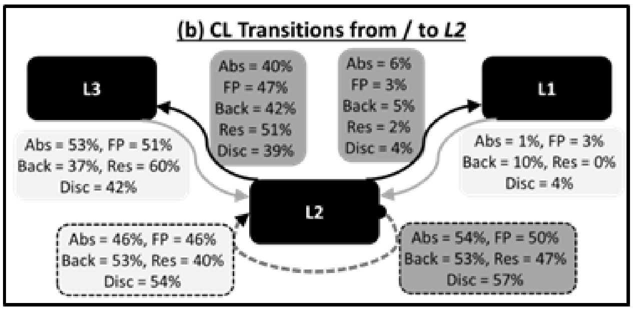


Fig. 4, continued. Continued to the next page.

Speculated events are completely absent in 28% of abstracts, which reinforces the claim that authors will only introduce uncertainty into abstracts where absolutely necessary. Of the remaining abstracts, a significant majority (58%) include the transition pattern *L3* → *L2*. These are the cases where authors deploy slight hedging on the analyses of their findings. Sometimes, this pattern is repeated 2 or 3 times, mostly when abstracts report on multiple sets of observations, each followed by its corresponding analysis. A small proportion of abstracts (5%) contain the pattern *L3* → *L2* → *L1*. As mentioned earlier, these are the cases where slightly hedged analyses are followed by bolder analyses, predictions or hypotheses, which can be a useful tool in helping to pique the reader's curiosity. Interestingly, a significant proportion of abstracts (14%) contain the transition pattern *L3* → *L1*, i.e., observations and confident analyses are followed directly by highly speculated analyses or hypotheses.

4 Full Papers

In this section we present a brief analysis of the meta-knowledge transitions observed in the *Background*, *Results*, and *Discussion* sections of the FP-MK corpus.

4.1 Knowledge Type (KT)

Figure 3 shows the summary of pair-wise transitions **from** and **to** adjacent events in the FP-MK corpus, according to *KT* categories. It in-

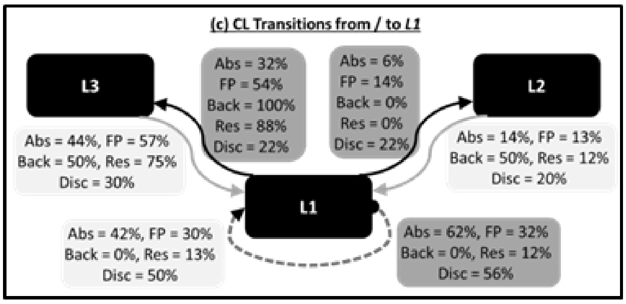


Fig. 4, continued.

cludes separate statistics for each of the main sections, as well as for the full papers as a whole.

Observation: Overall distributions of transitions from and to *Observation* in full papers are similar to those in abstracts. However, the reflexivity of *Observation* is slightly lower in full papers. This is partly because of the significantly higher proportion of transitions between *Observation* and *Analysis* in full papers. Full papers contain many more observations, most of which are subsequently further analysed. This kind of linking between observations and analyses is particularly frequent in the *Results* and *Discussion* sections. Full papers contain slightly fewer transitions from *Observation* to *Investigation*. This is mainly because the relative frequency of *Investigation* events is considerably lower in full papers than in abstracts.

Analysis: Full papers contain significantly more transitions from *Analysis* to *Fact*, especially in *Background* and *Discussion* sections. This is because the stringent size constraints imposed for abstracts are relaxed for the body of full papers, and thus authors have greater opportunity to relate their work to the state-of-the-art in their domain. The overall reflexivity of *Analysis* events is slightly less in full papers than in abstracts. This is despite the fact that the overall relative frequency of *Analysis* events in full papers is higher than in abstracts. This can be explained by the more complex interweaving of analytical statements with observations or facts that is often found in full papers, as evidenced by the much higher number of transitions from *Analysis* to *Observation* in full papers. Such patterns have particularly high frequency in the *Results* and *Discussion* sections of papers. Finally, full papers contain significantly fewer transitions from *Analysis* to *Investigation*. This is mainly because *Investigation* events rarely occur in some sections of full papers, whereas many abstracts contain a small number of *Investigation* events.

Investigation: Overall reflexivity of *Investigation* events in full papers is significantly less than in abstracts, due to a lower relative frequency of *Investigation* events in full papers. Full papers contain significantly higher numbers of transitions from *Investigation* events to *Method* events. Interestingly, almost all of these transitions are in the *Results* sections. This is probably due to the need to explain how particular aspects of the investigation were carried out by applying particular experimental methods. A similar percentage of transitions can be observed between *Method* and *Observation* events in the *Results* sections, showing that the next step is often to describe how the use of the method led to particular experimental observations. Full papers also contain slightly more transitions from *Investigation* events to *Analysis* events, especially in *Discussion* sections, where a direct link is made between the investigations undertaken and the findings resulting from them.

Fact: Overall distributions are similar to abstracts, with one minor difference: full papers contain more transitions from *Fact* to *Method*, especially in *Background* and *Discussion* sections. This is mainly because sometimes, authors make a direct link between background facts and the experimental methods used, omitting the intermediary link to investigations. This is especially the case when authors have already mentioned the investigations earlier in the text.

Method: We found no significant differences in the distribution of *Method* events in full papers and abstracts. This is partly due to the scarcity of *Method* events (in both GENIA-MK and FP-MK corpora) caused by the definition of bio-event used to annotate these corpora, which excludes many method descriptions from event annotation.

4.2 *Certainty Level (CL)*

L3: The distributions of transitions from/to *L3* events in full papers are similar to those in abstracts, except for one main difference: Full papers contain slightly more transitions from *L3* to *L2* events. This is due to more detailed analytical discussion often found in full papers. Moreover, unlike in abstracts, where the main aim is to try to sell the research results, the body of the paper provides greater opportunity for analysis and discussion. The percentage of *L3* to *L2* transitions is highest in the *Results* sections of the full papers. Authors may be confident about some of their results, but not so confident about others. Fewer such transitions are found in the *Discussion* section, suggesting that authors take a more confident tone in analysing their most definite results, in order to convince the reader of the reliability of their conclusions.

L2: Full papers contain slightly more transitions from *L2* to *L3* events. This is mainly due to the more frequent occurrence of contiguous observation-analysis transitions. Full papers contain significantly fewer transitions from *L2* to *L1* events. As mentioned above, such transitions are often made in abstracts for increased effect or impact. If too many bold or controversial statements are made in the body of the paper, readers may question the integrity of the study.

L1: Overall reflexivity of *L1* events is much lower in full papers than in abstracts. Although the relative frequency of L1 events is higher in full papers, they are more thinly spread out. The greater the number of highly speculative events that occur in sequence, the more wary the reader is likely to become.

5 Conclusion

In this paper, we have investigated discourse patterns that occur in biomedical abstracts and full papers. In contrast to previous work on discourse structure, our analysis was conducted at the level of bio-events. We used the GENIA-MK corpus of abstracts and the FP-MK corpus of full paper to conduct our analyses. We examined a number of different types of discourse patterns, including patterns of pairwise transitions between events, considering *KT* and *CL* separately. Comparison of the results obtained for abstracts and full papers reveal that there are a number of subtle and significant differences in the patterns of local discourse-level shifts. For abstracts, we additionally considered extended transition paths. Whilst there are some clear patterns of *KT* and *CL* transitions in abstracts, these are by no means standard. Furthermore, while most abstracts follow a generic model of rhetoric/information moves, authors often skip certain moves, assuming that the reader is already familiar with the context.

As future work, we intend to broaden the scope of our study to incorporate different types of events and additional meta-knowledge dimensions across different domains. We also plan to investigate transition patterns within each section of full papers. Furthermore, with the help of the BioDRB corpus, we intend to investigate correlations between particular types of discourse relations and the meta-knowledge values of the events that occur within the argument text spans of these relations.

ACKNOWLEDGMENTS The work described in this paper has been funded by the MetaNet4U project (ICT PSP Programme, Grant Agreement: No 270893).

References

1. Sandor, Å., de Waard, A.: Identifying Claimed Knowledge Updates in Biomedical Research Articles. Proceedings of the Workshop on Detecting Structure in Scholarly Discourse (DSSD), 2012, 10–17.
2. Oda, K., Kim, J.-D., Ohta, T., Okanohara, D., Matsuzaki, T., Tateisi, Y., Tsujii, J.i.: New challenges for text mining: mapping between text and manually curated pathways. BMC Bioinformatics 9, 2008, S5.
3. Yeh, A.S., Hirschman, L., Morgan, A.A.: Evaluation of text data mining for database curation: lessons learned from the KDD Challenge Cup. Bioinformatics 19, 2003, i331-i339.
4. Teufel, S., Carletta, J., Moens, M.: An annotation scheme for discourse-level argumentation in research articles. Proceedings of EACL, 1999, 110–117.
5. Mizuta, Y., Korhonen, A., Mullen, T., Collier, N.: Zone analysis in biology articles as a basis for information extraction. International Journal of Medical Informatics 75, 2006, 468-487.
6. Wilbur, W.J., Rzhetsky, A., Shatkay, H.: New directions in biomedical text annotations: definitions, guidelines and corpus construction. BMC Bioinformatics 7, 2006, 356.
7. Marcu, D., Echihabi, A.: An unsupervised approach to recognizing discourse relations. Proceedings of ACL. Association for Computational Linguistics, 2002, 368–375.
8. Swales, J.: Genre Analysis: English in Academic and Research Settings: Cambridge Applied Linguistics. Cambridge University Press, 1990.
9. Teufel, S.: Argumentative Zoning. University of Edinburgh, 1999.
10. Marcus, M.P., Santorini, B., Marcinkiewicz, M.A.: Building a large annotated corpus of English: The Penn Treebank. Computational Linguistics 19, 1994, 313–330.
11. Carlson, L., Marcu, D., Okurowski, M.E.: Building a Discourse-Tagged Corpus in the Framework of Rhetorical Structure Theory. Current and New Directions in Discourse and Dialogue. In: Kuppevelt, J., Smith, R.W. (eds.), Vol. 22. Springer Netherlands, 2003, 85–112.

12. Mann, W.C., Thompson, S.A.: Rhetorical structure theory: Toward a functional theory of text organization. Text 8, 1988, 243–281.
13. Prasad, R., Dinesh, N., Lee, A., Miltsakaki, E., Robaldo, L., Joshi, A., Webber, B.: The Penn Discourse TreeBank 2.0. Proceedings of the 6th International Conference on Language Resources and Evaluation (LREC), 2008, 2961–2968.
14. Prasad, R., McRoy, S., Frid, N., Joshi, A., Yu, H.: The biomedical discourse relation bank. BMC Bioinformatics 12, 2011, 188.
15. Ananiadou, S., Pyysalo, S., Tsujii, J., Kell, D.B.: Event extraction for systems biology by text mining the literature. Trends Biotechnol. 28, 2010, 381–390.
16. Kim, J.-D., Ohta, T., Tsujii, J.: Corpus annotation for mining biomedical events from literature. BMC Bioinformatics 9, 2008.
17. Thompson, P., Iqbal, S., McNaught, J., Ananiadou, S.: Construction of an annotated corpus to support biomedical information extraction. BMC Bioinformatics 10, 2009, 349.
18. Pyysalo, S., Ohta, T., Miwa, M., Cho, H.-C., Tsujii, J.i., Ananiadou, S.: Event extraction across multiple levels of biological organization. Bioinformatics 28, 2012, i575–i581.
19. Kim, J.-D., Pyysalo, S., Nedellec, C., Ananiadou, S., Tsujii, J. (eds.): Selected Articles from the BioNLP Shared Task 2011, Vol. 13. BMC Bioinformatics, 2012.
20. Nawaz, R., Thompson, P., McNaught, J., Ananiadou, S.: Meta-Knowledge Annotation of Bio-Events. Proceedings of LREC 2010, 2010, 2498–2507.
21. Thompson, P., Nawaz, R., McNaught, J., Ananiadou, S.: Enriching a biomedical event corpus with meta-knowledge annotation. BMC Bioinformatics 12, 2011, 393.
22. Nawaz, R., Thompson, P., Ananiadou, S.: Meta-Knowledge Annotation at the Event Level: Comparison between Abstracts and Full Papers. Proceedings of the Third LREC Workshop on Building and Evaluating Resources for Biomedical Text Mining (BioTxtM 2012), 2012, 24-21.
23. Miwa, M., Thompson, P., McNaught, J., Kell, D.B., Ananiadou, S.: Extracting semantically enriched events from biomedical literature. BMC Bioinformatics 13, 2012, 108.
24. Nawaz, R., Thompson, P., Ananiadou, S.: Identification of Manner in Bio-Events. Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC 2012), 2012.
25. Liakata, M., Thompson, P., de Waard, A., Nawaz, R., Maat, H.P., Ananiadou, S.: A Three-Way Perspective on Scientific Discourse Annotation for Knowledge Extraction. Proceedings of the Workshop on Detecting Structure in Scholorly Discourse (DSSD), 2012, 37–46.
26. Sauri, R., Pustejovsky, J.: FactBank: A Corpus Annotated with Event Factuality. Language Resources and Evaluation 43, 2009, 227-268.
27. Palmer, M., Gildea, D., Kingsbury, P.: The Proposition Bank: An Annotated Corpus of Semantic Roles. Computational Linguistics 31, 2005, 71–106.

28. Ruppenhofer, J., Ellsworth, M., Petruck, M., Johnson, C., Scheffczyk, J.: FrameNet II: Extended Theory and Practice, 2010.
29. Swales, J.: Genre Analysis: English in Academic and Research Settings. Cambridge University Press, 1990.

RAHEEL NAWAZ
NATIONAL CENTRE FOR TEXT MINING,
MANCHESTER INTERDISCIPLINARY BIOCENTRE,
UNIVERSITY OF MANCHESTER,
131 PRINCESS STREET, MANCHESTER, M1 7DN, UK
E-MAIL: <RAHEEL.NAWAZ@CS.MAN.AC.UK>

PAUL THOMPSON
NATIONAL CENTRE FOR TEXT MINING,
MANCHESTER INTERDISCIPLINARY BIOCENTRE,
UNIVERSITY OF MANCHESTER,
131 PRINCESS STREET, MANCHESTER, M1 7DN, UK
E-MAIL: <PAUL.THOMPSON@MANCHESTER.AC.UK>

SOPHIA ANANIADOU
NATIONAL CENTRE FOR TEXT MINING,
MANCHESTER INTERDISCIPLINARY BIOCENTRE,
UNIVERSITY OF MANCHESTER,
131 PRINCESS STREET, MANCHESTER, M1 7DN, UK
E-MAIL: <SOPHIA.ANANIADOU@MANCHESTER.AC.UK>

IJCLA VOL. 4, NO. 2, JUL-DEC 2013, PP. 121–133
RECEIVED 05/12/12 ACCEPTED 04/03/13 FINAL 09/03/13

Medical Event Extraction using Frame Semantics — Challenges and Opportunities

DIMITRIOS KOKKINAKIS

University of Gothenburg, Sweden

ABSTRACT

The aim of this paper is to present some findings from a study into how a large scale semantic resource, FrameNet, can be applied for event extraction in the (Swedish) biomedical domain. Combining lexical resources with domain specific knowledge provide a powerful modeling mechanism that can be utilized for event extraction and other advanced text mining-related activities. The results, from developing a rule-based approach, showed that only small discrepancies and omissions were found between the semantic descriptions, the corpus data examined and the domain-specific semantics provided by SNOMED CT (medical terminology), NPL (medicinal products) and various semi-automatically developed clue lists (e. g., domain-related abbreviations). Although the described experiment is only based on four different domain-specific frames, the methodology is extendable to the rest ones and there is much room for improvements, for instance by combining rule-based with machine learning techniques, and using more advanced syntactic representations.

KEYWORDS: Event extraction; frame semantics; semantic arguments; FrameNet.

1 Introduction

Natural language understanding (NLU), is a subtopic and a long-term goal for Natural Language Processing (NLP), which aims to enable computers to derive meaning from natural language input. NLU systems require a semantic theory to guide the comprehension of any text and at the same time a suitable framework for representing lexical knowledge, preferably linked to domain ontologies and terminologies. In such a context, a semantic-oriented framework could play a vital role for alleviating the extraction of complex semantic relations and, often pre-specified, simple or composite events. Event-based, or event-template information extraction have been initiated by and explored in the MUC-4 extraction task [1]. Since then, extraction and labeling of events has also attracted attention in various activities (e.g. in the SEMEVAL framework [2] and the BioNLP shared tasks [3]). In recent years, algorithms are also developed that try to learn instead template structures automatically from raw text; *cf.* [4]. Here, we are interested in biomedical event extraction, which refers to the task of extracting descriptions of actions and relations among one or more entities from the biomedical literature.

Mining such complex relations and events has gained a growing attention in this domain; [3, 5, 6] and for several reasons. Mainly due to the existence of a publication volume that increases at an exponential level, the availability of mature NLP tools for biomedical text analysis, large lexical/terminological/ontological resources, and various manually annotated samples with semantic information. All these factors have resulted in an explosion of event-related research in the domain (*cf.* <http://nactem.ac.uk/genia/>, <https://www.i2b2.org/>). Semantically driven literature analysis and literature-based knowledge discovery provide a lot of challenging research topics and a paradigm shift is taking place in the biomedical domain, from relation models in information extraction research to more expressive event models, ***cf.*** [7].

Our approach is closely related to information extraction (IE), a technology that has a direct correlation with frame-like structures as described in the FrameNet. Templates in the context of IE are frame-like structures with slots representing event information. Most event-based IE approaches are designed to identify role fillers that appear as arguments to event verbs or nouns, either explicitly via syntactic relations or implicitly via proximity. In this paper we argue that frame semantics is such a framework that can facilitate the development of text understanding and as such can be used as a backbone to NLU systems. We present results from experiments using domain-specific FrameNet extensions for the automated analysis of meaning in Swedish medical texts. With this approach we aim to develop and apply automatic event extraction in the Swedish medical domain in a large scale and in the long run, we are particularly interested in developing a set of tools to support health care professionals and researchers to rapidly identify, aggregate and semantically exploit relevant information in large textual repositories.

2 Theoretical Background

The FrameNet approach is based on the linguistic theory of frame semantics [8] supported by corpus evidence. A semantic frame is a script-like structure of concepts which are linked to the meanings of linguistic units and associated with a specific event, situation or state. Each frame identifies a set of frame elements, which are frame specific semantic participants and roles/arguments (both core and non-core ones). Furthermore, roles may be expressed overtly, left unexpressed or not explicitly linked to the frame via linguistic conventions (null instantiations). In this work, we only deal with the first type of such roles. FrameNet documents the range of semantic and syntactic combinatory possibilities of frame evoking lexical units (LU), phrases and clauses by abstracting away from syntactic differences. A LU can evoke a frame, and its syntactic dependents can fill the frame element slots, in turn, the various semantic types constrain the types of frame element fillers. Since a LU is the pairing of a word with a meaning, each sense of a polysemous word belongs to a different semantic frame, Moreover, since a single frame element can have different grammatical realizations it can enhance the investigation of combinatorial possibilities more precisely than other standard lexical resources such as WordNet.

2.1 *The Swedish FrameNet*

The Swedish FrameNet (SweFN++) is a lexical resource under active development, based on the English version of FrameNet constructed by the Berkeley research group. The SweFN++ is available as a free resource at <http://spraakbanken.gu.se/swefn/>. Most of the SweFN frames and frame names correspond to the English ones, with some exceptions, as to the selection of frame elements including definitions and internal relations. Compared to the Berkeley FrameNet, SweFN++ is expanded with information about the domain of the frames, at present the medical and the art domain. Since frame classification is based on general-domain frame semantics, several efforts have been described to domain adaptations even for English [9, 10].

As of November 2012, the SweFN++ covered 754 frames with around 24,000 lexical units, while 30 frames are marked as medically-oriented; [11]. The lexical units are gathered from SALDO, a free Swedish electronic association lexicon [12]. FN facilitates modeling the mapping of form and meaning within these structures in the medical discourse through manual annotation of example sentences and automatic summarization of the resulting annotations. Some of the medical frames in SweFN include: *Addiction*; *Cure*; *Recovery*; *Experience_bodily_harm*; *Falling_Ill*; *Administration_of_medication* etc. For instance, the Cure frame describes a situation involving a number of core roles such as: *Affliction*, *Healer*, *Medication*, *Patient* etc., and a number of non-core roles such as *Degree*, *Manner* and *Time*, and it is evoked by lexical units such as to *cure*, to *heal*, *surgery*, and to *treat*. The word in bold face below evokes the *Cure* frame: "`[Steloperation av fotleden]-TREATMENT {`**`lindrar}`**`-CURE [smärta]-AFFLICTION [väl]-MANNER men medför en del komplikationer`" (litt. 'Lumbar fusion operation of the ankle reduces pain well, but entails some complications').

3 Experimental Setting

Our approach uses the annotation results produced from the application of adapted entity and terminology taggers; as a semantic theory the use of specifically designed medical frames, with associated manually annotated textual samples, and, finally, various manually developed frame related regular expression patterns. The domain-specific medical frames we have been using are: *Administration_of_medication*, with core frame elements such as *Drug*, *Patient* and *Medic* (112), *Medical_Treatment*, with core frame elements such as *Treatment*, *Affliction* and *Patient* (102), *Cure*, with core frame elements such as *Healer*, *Affliction* and *Body_Part* (115) and *Falling_Ill*, with core frame elements such as *Patient*, *Symptom* and *Ailment* (116); the figure in parenthesis refers to the number of manually annotated sentences, randomly extracted from a large available Swedish biomedical corpora [13]. All annotated samples are available from the following addresses: http://demo.spraakdata.gu.se/brat/#/[sweFNCure_dk; sweFNMed-Treatment_dk; sweFNFallingIll_dk; sweFNAdminOfMed_dk].

3.1 *Relevant Resources*

We have been using a number of relevant resources (textual, terminological, etc.) for modeling pattern matching rules, i.e. complex regular expressions. Some of the most important resources have been used for both extracting relevant text samples and also aiding the recognition of relevant frame elements in the samples. The main source for medical terminology has been the Swedish Nomenclature of Medicine, Clinical Terms (SNOMED CT), since it is the largest available source of medical terminology in Swedish, approx. 300,000 terms. Medication names are provided by the National Repository for Medicinal Products (NPL, <http://www.lakemedelsverket.se>) which is the official Swedish product registry for drugs, approx. 12,000 terms.

Every product in this registry contains information on its substances, names, dosages, producers and classifications like prescription and Anatomical Therapeutic Chemical codes (ATC). Lists of semi-automatic acquired drug / substance / disease lexicon extensions (e.g. generic expressions of drugs and diseases, misspellings etc.); lists of key words (e.g. drug forms [pill, tablet, capsule], drug administration paths [intravenous, intravesical, subcutaneous], volume units [mg, mcg, IE, mmol] and various abbreviations and variants [iv, i.v., im, i.m. sc, s.c., po, p.o., vb, T]). Finally, important pieces of information are also obtained by the application of named entity recognition, which identifies and annotates very important frame elements, particularly time expressions, various types of numerical information (such as dosage and frequency) and some terminology (such as lists of non-official drug names).

3.2 *Method*

As a method we apply a rather simple, rule-based approach (which can be used as a baseline for future work using other techniques) by performing three major steps. (i) *pre-processing*, that is selecting a relevant sample of sentences for each frame using trigger words (i.e. relevant LUs) for both manual annotation and pattern development and evaluation, (ii) *main processing*, which includes terminology, named entity and key word/text segment identification, (iii) *post-processing*, e.g., modeling observed frame element patterns as rules (regular expressions). All steps are applied at the sentence level, i.e. no coherent, larger text fragments are used. First, we manually annotated the sentence samples with all possible frame elements. Through the manual analysis of the annotated examples we could obtain a rather good understanding of how the examined medical events can be expressed in the data. This way we can model various rules for the task and also have annotated data for future planned supervised learning extensions. During processing, we first start by identifying and annotating the terminology (e.g. SNOMED CT terms and NPL drug names) or drug name classes (e.g., antibiotics). For the main processing step we apply named entity recognition which identifies and annotates relevant frame elements such as time expressions, various important numerical entity information types, named entities such as person and location and also non-official terminology.

These annotations are important since they are both required by the frames and appear regularly in the context of the medical frames. A number of lexical rules, as previously described, based on e.g. lists of administration paths for drug admission etc., implemented as regular expressions are applied for the recognition and annotation of relevant frame elements. Using as a guidance the order of the extracted element patterns from the annotated sample, we model those as rules. For instance, the most frequent frame element pattern in the *Administration_of_Medication* frame (10 occurrences; 20 combined with other elements) is "<Drug_name> <Drug_strength> <Frequency>", and in the *Falling_Ill* frame (22 occurrences; 46 combined with other elements) is "<Patient> <Ailment>".

An annotated example sentence with named entities, from the *Administration_of_Medication* frame, is shown below, the XML-like labels should be self-explanatory. Here, the entity tagger annotates occurrences of time ("TIMEX/TME"); frequency ("NUMEX/FRQ") and dosage ("NUMEX/DSG"): `Åtta patienter erhöll Recormon före operationen, i dosering 2 000 IE subkutant tre gånger per vecka under tre veckor` (litt. 'Eight patients received Recormon before surgery, dosage 2000 IU subcutaneously three times per week for three weeks') is annotated as `Åtta patienter erhöll Recormon före operationen, i dosering <NUMEX TYPE="MSR" SBT="DSG">2 000 IE</NUMEX> subkutant <NUMEX TYPE="MSR" SBT="FRQ">tre gånger per vecka </NUMEX> <TIMEX TYPE="TME" SBT="DAT"> under tre veckor</TIMEX>`. All labels were normalized to their frame element names at a later stage. For instance, the following example from the *Administration_of_Medication* frame, illustrates an example with normalized frame element labels: `Lugnande besked, rec <Drug_name>Tradil</Drug_name> <Drug_strength> 400 mg</Drug_strength> <Frequency>1 x 1-2 </Frequency>` (litt. 'Reassurance, rec Seractil 400 mg 1 x 1-2').

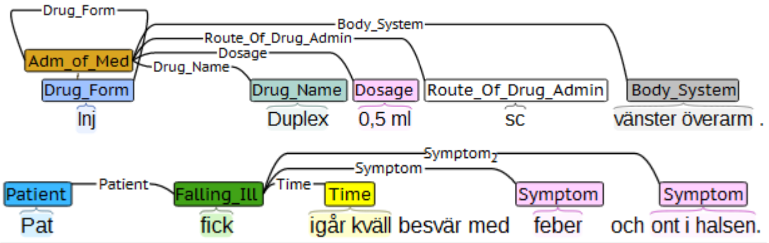


Fig. 1. Examples of manually annotated data with the frames *Administration_of_Medication* (top) and the *Falling_Ill* (bottom) using the *brat* annotation tool [14].

4 Results and Discussion

Table 1 shows the evaluation results (complete match) for the top 4 frame elements (most occurrences in a test set of 30x4 sentences) for the four examined domain frames. Some of the no-core frame elements could not be found in the sample, while some had very few occurrences and this is the reason we chose not to formally evaluate all of those at this stage. This vertical level evaluation assess the extraction of each frame element individually. A number of problematic issues still remain. For instance, certain elements are difficult to capture using regular expressions, such as <Purpose>, <Outcome> and <Circumstance>. These seem the most problematic since these element shows great variability and expressed by common language patterns. Perhaps syntactic parsing needs to be exploited in such cases because these elements are often described by lengthy, complex noun or prepositional phrases and clauses.

For instance, the following example shows a prepositional phrase complex with four prepositions (in bold face): <Circumstance> **Vid** klart skyldig blindtarmsinflammation **av** varierande grad upp till kraftigare inflammation **med** tecken **på** vävnadsdöd i blindtarmen </Circumstance> administreras antibiotika Tienam 0,5 g x 3 (litt. 'In clear-cut case appendicitis of varying degree up to stronger inflammation with signs of necrosis in the cecum antibiotic Tienam 0.5 g x 3 is administered'). Another problematic aspect is observed for many cases where there is an ellipsis, that is, clauses where an overt trigger word is missing (often a predicate belonging to the frame). For instance, the following example shows such an ellipsis, lack of an overt trigger, a verb, in the last clause marked in italic: Av journalblad framgår att han behandlats med digitalis , såväl i injektion som per os, *samt med kinidin tabletter*. (litt. Of the record sheet it is shown that he has been treated with digitalis, both injection and per os, and with quinidine tablets.)

Table 1. Evaluation of the most frequent frame elements in the test sample.

Frame	Frame elements			
Admin. of_Medic.	Drug_Name 92,6%(Pr) 81,2%(R)	Dosage 96%(Pr) 90,1%(R)	Frequency 98,7%(Pr) 91,9%(R)	Route_Of_Drug_Admin 100%(Pr) 97,1%(R)
Cure	Affliction 94%(Pr) 92,9%(R)	Treatment 83,1%(Pr) 79,2%(R)	Patient 100%(Pr) 100%(R)	Medication 94%(Pr) 89,2%(R)
Falling_Ill	Patient 100%(Pr) 95%(R)	Ailment 88,9%(Pr) 91,1%(R)	Symptom 78,9%(Pr) 83.4%(R)	Time 100%(Pr) 100%(R)
Medical _Treatment	Patient 100%(Pr) 100%(R)	Affliction 93,2%(Pr) 91%(R)	Medication 97,9%(Pr) 95%(R)	Time 100%(Pr) 100%(R)

In Table 1, Precision measures the amount of elements correctly labeled, out of the total number of all elements labeled by the rules; while Recall measures the amount of elements correctly labeled given all of the possible elements in the sample. The evaluation results are based on sentences for each frame that were annotated separately from the annotated sample used for the creation of the pattern matching rules (these sentences were annotated and evaluated by the author). Nevertheless, it should have been advantageous if (trained) experts, e.g. physicians, could annotate the test data but that was prohibitive at the moment, but will be considered in future, larger scale evaluations and method combinations.

As previously discussed, some of the frame elements could not be found in the annotated samples, while some had very few occurrences and were not formally evaluated, for instance the element *Place* in the *Falling_Ill* frame. Moreover, the manual annotation gave us the opportunity to revise some of the frame elements and in a revised version of the frames in SweFN++, some of the domain frames will be divided in two. Thus in order to get even more accurate and precise semantics (arguments) some frames would require more *specialization*. For instance, the *Administration_of_medication* would be required to be divided between *Administration_of_medication_conveyance* (where the procedures that describe the administration of medicine will be the focus of the frame; e.g. `Normalt ska en salva eller kräm strykas på tunt`; litt. "Normally, an ointment or cream will be thinly applied") and *Administration_of_medication_specification* (where the focus should be on the specifications concerning administration of medicines; e.g. `Tegretol 20 mg/ml, 30 ml x 1`).

5 Conclusion and Future Work

We have presented a set of experiments using a rule-based approach on automatic semantic role labeling, and in particular event-based information extraction, using frame semantics modeled in the Swedish FrameNet. We have investigated the use and efficacy of a rule-based approach for the recognition and labeling of the semantic elements, on a specialized textual domain, namely biomedicine. So far we have been working with four different frames and experimenting with simple pattern matching approaches in order to use as a baseline for future experiments. The driving force for the experiments is the theory of frame semantics, which allows us to work with a holistic and detailed semantic event description than it has been previously reported in similar tasks or in efforts using, for instance, most traditional methods based on relation extraction. Moreover, event extraction is more complicated and challenging than relation extraction since events usually have internal structure involving several entities as participants allowing a detailed representation of more complex statements.

Due to the small amount of labeled data, we have not yet attempted to apply a machine learning approach, since such as classifier would suffer from feature sparsity. However, annotating sentences is very time-consuming and we will thus have to live with small training sets for the near future. Still, this problem can be addressed in several ways; for instance through the use of cross-frame label generalization and by adding cluster-based features. In a similar fashion, Johansson et al. [15] have shown that such methods result in clear performance improvements. This way, traditional, lexicalized approaches may lead into other research paradigms, such as semi-supervised approaches [16] and the inclusion of automatically produced training data [17]. In the near future we intend to investigate the validity of the medical frames by manually annotating authentic samples for all available medical frames and also combine the pattern-based approaches with supervised learning for automatic extraction and labeling of frame elements. Note, however that we have observed that in some cases/frames, such as *Administration_of_Medication,* simple means implemented as regular expressions are enough for accurate identification of frame elements, since such a frame contains a plethora of numerical information and domain-specific abbreviations and acronyms that require less advanced techniques in order to obtain good coverage. In other cases, such as in the *Cure* frame, other means seem more appropriate, such as syntactic parsing.

Event recognition at the moment is performed at a sentence level using a nearly homogeneous corpus of biomedical Swedish and also overuse of trigger words. One of the future challenges is of course to treat the problem of event detection as a classification one where one could strive to rely less on the presence of such trigger words. On the other side rule-based methods on domain-specific events and frames with a limited set of vocabulary (lexical units) can be as efficient or even outperform classification accuracy. Moreover, it has been shown that the most effective classification approach is dependent on the target event type [18]. Events that can be described by a large set of lexical units (many synonymous, near-synonymous etc.) are more suitable for training purposes and thus more efficient using a classification approach, while for events using a limited set of vocabulary a triggers' based classification system produces better results. Therefore, in the future, we plan to compare which technique is most appropriate for which type of frame.

ACKNOWLEDGMENTS This work was supported by the Centre for Language Technology, http://clt.gu.se/, at the University of Gothenburg and the Swedish Research Council project: *Swedish FramNet++ — a lexical infrastructure for Swedish language technology* (2011–13).

References

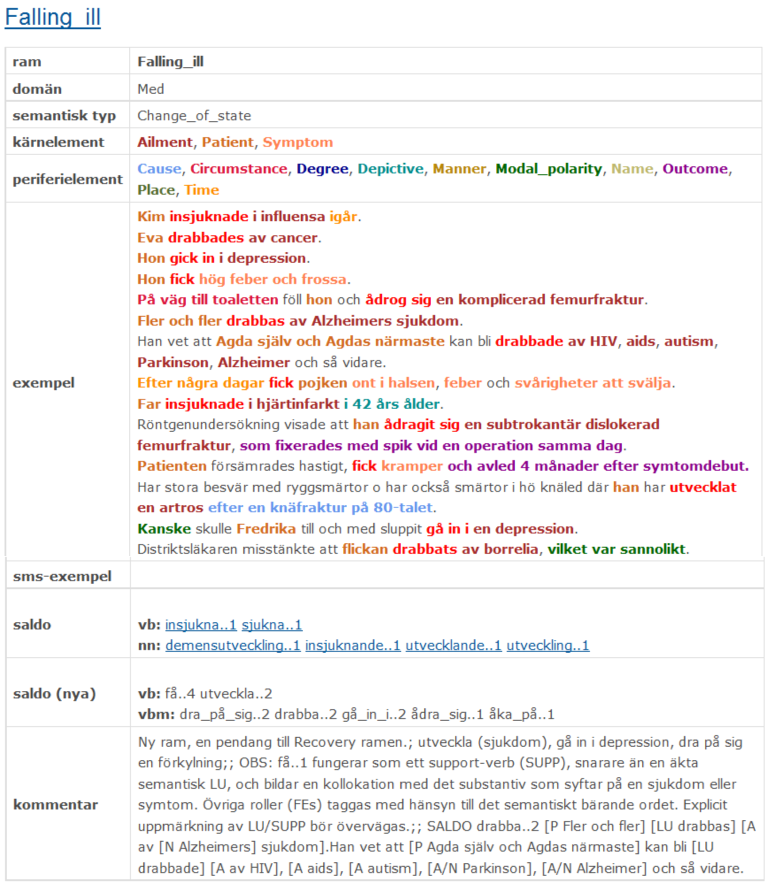
1. Rau, L., Krupka, G., Jacobs, P., Sider, I., Childs, L.: MUC-4 test results and analysis. 4th Message Understanding Conf. (MUC-4), 1992.
2. Ruppenhofer, J. et al.: SemEval-2010 Task 10: Linking Events and Their Participants in Discourse. The NAACL-HLT 2009 Workshop on Semantic Evaluations: Recent Achievements and Future Directions. Colorado, USA, 2010.

3. Kim, JD., Pyysalo, S., Ohta, T., Bossy, R., Tsujii, J: Overview of BioNLP Shared Task 2011. Proceedings of the BioNLP 2011 Workshop Companion Volume for Shared Task 1–6, 2011.
4. Chambers, N., Jurafsky, D.: Template-based information extraction without the templates. Proc of the 49th Annual Meeting of ACL: HLT. 976–986. Oregon, USA, 2011.
5. Ananiadou, S., Pyysalo, S., Tsujii, J., Kell, DB.: Event extraction for systems biology by text mining the literature. Trends Biotechnol. 28(7):381–90, 2010.
6. Miwa, M., Thompson, P., McNaught, J., Kell, DB., Ananiadou, S.: Extracting semantically enriched events from biomedical literature. BMC Bioinformatics, 13:108, 2012.
7. Björne, J., Ginter, F., Pyysalo, S., Tsujii, J., Salakoski, T.: Complex event extraction at PubMed scale. Bioinformatics 15;26(12):i382-90, 2010.
8. Fillmore, CJ., Johnson, CR., Petruck, MRL.: Background to FrameNet. J of Lexicography. 16(3), 2003.
9. Dolbey, A., Ellsworth, M., Scheffczyk, J.: BioFrameNet: A Domain-Specific FrameNet Extension with Links to Biomedical Ontologies. 2nd Workshop on Formal Biomed. Knowledge Repres. (KR-MED). Baltimore, USA, 2006.
10. Tan, H., Kaliyaperumal, R., Benis, N.: Ontology-driven Construction of Corpus with Frame Semantics Annotations. Proceedings of the 13th International Conference on Intelligent Text Processing and Computational Linguistics (CICLing). New Delhi, India, 2012.
11. Borin, L., Dannélls, D., Forsberg, M., Toporowska Gronostaj, M., Kokkinakis, D.: The past meets the present in the Swedish FrameNet++. In Proceedings of EURALEX, 2010.
12. Borin, L.: Med Zipf mot framtiden - en integrerad lexikonresurs för svensk språkteknologi. LexicoNordica, 17 (In Swedish), 2010.
13. Kokkinakis, D.: The Journal of the Swedish Medical Association - a Corpus Resource for Biomedical Text Mining in Swedish. The Third Workshop on Building and Evaluating Resources for Biomedical Text Mining (BioTxtM), an LREC Workshop. Turkey, 2012.
14. Stenetorp, P., Pyysalo, S., Topic, G., Ohta, T., Ananiadou, S., Tsujii, J.: brat: a Web-based Tool for NLP-Assisted Text Annotation. Proc of the Eur. ACL. 102–107. Avignon, France, 2012.
15. Johansson, R., Friberg Heppin, K., Kokkinakis, D.: Semantic Role Labeling with the Swedish FrameNet. Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC). 3697–3700. Istanbul, Turkey, 2012.
16. Fürstenau, H., Lapata, M.: Semisupervised semantic role labeling. In Proceedings of the 12th Conference of the European Chapter of the ACL (EACL). 220–228, Athens, Greece, 2009.
17. Johansson, R., Nugues, P.: A FrameNet-based semantic role labeler for Swedish. In Proceedings of Coling/ACL, Sydney, Australia, 2006, 436–443.

18. Naughton, M., Stokes, N., Carthy, J.: Sentence-Level Event Classification in Unstructured Texts. Tech. Report UCD-CSI-2008-07. School of Computer Science and Informatics, University College Dublin, Ireland, 2008.

Appendix

The *Falling_Ill* frame. Domain [*domän*]: Medicine; Semantic Type [*semantisk typ*]: Change_of_State; Core Elements [*kärnelement*]; Non-Core Elements [*periferielement*]; Examples [*exempel*]; Lexical units [*saldo*]; Comments [*kommentar*].



Falling_ill

ram	**Falling_ill**
domän	Med
semantisk typ	Change_of_state
kärnelement	Ailment, Patient, Symptom
periferielement	Cause, Circumstance, Degree, Depictive, Manner, Modal_polarity, Name, Outcome, Place, Time
exempel	Kim insjuknade i influensa igår. Eva drabbades av cancer. Hon gick in i depression. Hon fick hög feber och frossa. På väg till toaletten föll hon och ådrog sig en komplicerad femurfraktur. Fler och fler drabbas av Alzheimers sjukdom. Han vet att Agda själv och Agdas närmaste kan bli drabbade av HIV, aids, autism, Parkinson, Alzheimer och så vidare. Efter några dagar fick pojken ont i halsen, feber och svårigheter att svälja. Far insjuknade i hjärtinfarkt i 42 års ålder. Röntgenundersökning visade att han ådragit sig en subtrokantär dislokerad femurfraktur, som fixerades med spik vid en operation samma dag. Patienten försämrades hastigt, fick kramper och avled 4 månader efter symtomdebut. Har stora besvär med ryggsmärtor o har också smärtor i hö knäled där han har utvecklat en artros efter en knäfraktur på 80-talet. Kanske skulle Fredrika till och med sluppit gå in i en depression. Distriktsläkaren misstänkte att flickan drabbats av borrelia, vilket var sannolikt.
sms-exempel	
saldo	**vb:** insjukna..1 sjukna..1 **nn:** demensutveckling..1 insjuknande..1 utvecklande..1 utveckling..1
saldo (nya)	**vb:** få..4 utveckla..2 **vbm:** dra_på_sig..2 drabba..2 gå_in_i..2 ådra_sig..1 åka_på..1
kommentar	Ny ram, en pendang till Recovery ramen.; utveckla (sjukdom), gå in i depression, dra på sig en förkylning;; OBS: få..1 fungerar som ett support-verb (SUPP), snarare än en äkta semantisk LU, och bildar en kollokation med det substantiv som syftar på en sjukdom eller symtom. Övriga roller (FEs) taggas med hänsyn till det semantiskt bärande ordet. Explicit uppmärkning av LU/SUPP bör övervägas.;; SALDO drabba..2 [P Fler och fler] [LU drabbas] [A av [N Alzheimers] sjukdom].Han vet att [P Agda själv och Agdas närmaste] kan bli [LU drabbade] [A av HIV], [A aids], [A autism], [A/N Parkinson], [A/N Alzheimer] och så vidare.

DIMITRIOS KOKKINAKIS
CENTRE FOR LANGUAGE TECHNOLOGY
AND THE SWEDISH LANGUAGE BANK,
UNIVERSITY OF GOTHENBURG,
SWEDEN
E-MAIL: <DIMITRIOS.KOKKINAKIS@SVENSKA.GU.SE>

IJCLA VOL. 4, NO. 2, JUL-DEC 2013, PP. 135–150
RECEIVED 07/12/12 ACCEPTED 30/01/13 FINAL 10/03/13

Web Entity Detection for Semi-structured Text Data Records with Unlabeled Data

CHUNLIANG LU,[1] LIDONG BING,[1] WAI LAM,[1] KI CHAN,[2] AND YUAN GU[1]

[1] *The Chinese University of Hong Kong, Hong Kong*
[2] *Hong Kong University of Science and Technology, Hong Kong*

ABSTRACT

We propose a framework for named entity detection from Web content associated with semi-structured text data records, by exploiting the inherent structure via a transformation process facilitating collective detection. To learn the sequential classification model, our framework does not require training labels on the data records. Instead, we make use of existing named entity repositories such as DBpedia. We incorporate this external clue via distant supervision, by making use of the Generalized Expectation constraint. After that, a collective detection model based on logical inference is proposed to consider the consistency among potential named entities as well as header text. Extensive experiments have been conducted to evaluate the effectiveness of our proposed framework.

KEYWORDS: Named entity recognition, unsupervised machine learning, web mining.

1 INTRODUCTION

Entity detection is an important problem which has drawn much research efforts in the past decade. A lot of investigation has been done for detecting named entities from natural language texts or free texts such as [1, 2]. It can support a large number of applications such as improving the quality of question answering [3]. In this paper, we investigate the problem of detecting named entities from Web content associated with semi-structured or tabular text data records as shown in Fig. 1 and Fig. 2, without manually labeled data. Some existing methods on detection also make use of unlabeled data using weakly-supervised method such as [4] and semi-supervised method such as [5]. However, these existing methods cannot effectively handle the detection task from such kind of text data. Another limitation of these methods is that they still need some manually labeled data.

The first kind of Web content that we wish to handle is a list of semi-structured text data records called a *semi-structured record set* as exemplified in Fig. 1, which is taken from CICLing 2013 website. It is composed of a set of record information typically arranged as a list of records. Within a record, there are fields with possibly completely different formats. However, similar fields across records are formatted in a similar manner. Moreover, it is highly likely that named entities, if any, found in similar fields in different records belong to the same entity type. For example, the text field with a link under the photo from each record in Fig. 1 belongs to person names.

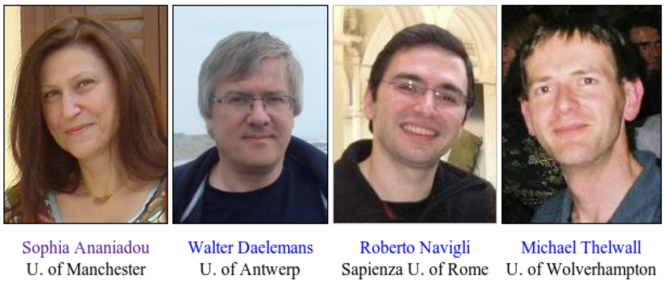
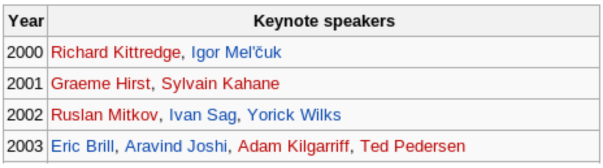


Fig. 1. An example of a semi-structured record set

The second kind of Web content is *tabular record set* as exemplified in Fig. 2. A tabular record set has a format similar to ordinary Web tables [6]. In general, multiple entities may exist in a single field. Most of fields under the same column share a common content type. A column may have a header text indicating the content of the column. For example, named entities found in the third column with header text “Keynote speakers” in Fig. 2 are person names.



Year	Keynote speakers
2000	Richard Kittredge, Igor Mel'čuk
2001	Graeme Hirst, Sylvain Kahane
2002	Ruslan Mitkov, Ivan Sag, Yorick Wilks
2003	Eric Brill, Aravind Joshi, Adam Kilgarriff, Ted Pedersen

Fig. 2. An example of a tabular record set

One common property for the above two content types is that they all have an inherent structure. For semi-structure record sets, each record can be segmented into fields. Corresponding fields with similar layout format in different records can be virtually aligned into a column. For tabular record sets, the structure can be readily obtained from HTML tags such as `<tr><td>`, with possible header text from `<th>` tags. The entities appeared in a particular column normally exhibit certain consistency between entities as well as header text, if any. This kind of structure information and possible column header text provide valuable guidance for the entity detection. We propose a framework that can exploit such underlying structure information via a transformation process facilitating collective detection. By incorporating existing named entity repositories such as DBpedia into the learning process via distant supervision, we do not require training labels on the data records. A collective detection model based on logical inference is proposed to consider the consistency among potential named entities as well as header text. Extensive experiments demonstrate the effectiveness of our framework.

2 Proposed Framework

2.1 *Overview*

Our framework focuses on two kinds of Web content mentioned above, namely, semi-structured record sets and tabular record sets. We transform these two kinds of record sets to a unified structure known as *structured field record lists*. A structured field record list consists of multiple records, with each record composed of multiple fields. A field is basically composed of text fragments possibly containing one or more, if any, named entities. Based on the layout format, corresponding fields in different records form a field column. A field column may optionally have a header text. We develop a component that is able to harvest semi-structured record sets from raw Web pages and transform the harvested record sets to structured field record lists based on the record field layout format. For tabular record sets, the detection and transformation are straightforward since we can directly examine HTML tags corresponding to tables.

The next component is to detect potential named entities from the generated structured field record lists. This component tackles the potential entity detection task for each record separately. To handle multiple entities possibly found in a field such as the records in Fig. 2, the detection is formulated as a sequence classification problem. Each record is tokenized as a token sequence and we aim to find the corresponding label sequence. We design labels based on the IOB format [7], and build a sequence classification model to predict the label for each token. To learn such a classification model, existing approaches rely on a large amount of training labels on the text data records. In contrast, our framework does not require training labels on the text data records. Instead, we leverage the existing large amount of labeled named entities from various external repositories such as DBpedia. We incorporate this external clue via distant supervision to guide the model learning. This paradigm is highly scalable in that it does not require tedious labeling effort.

After potential entities for each record are found as described above, the next component in our framework aims at taking advantage of the inherent structure information underlying the record list and considering the inter-relationships among records in the record list. One clue is that potential entities appeared in a particular field column of a record list generally share the same entity type. Another consideration is that some field columns may have header texts which can provide useful clues about the entity type of potential entities under those columns. A collective inference model is developed for incorporating all these clues based on logic paradigm. By exploiting such kind of structure information, better entity detection performance can be achieved.

2.2 *Identifying and Transforming Semi-structured Record Sets*

We first identify semi-structured record sets from Web page content. Then we conduct layout format driven alignment among the records in a record set resulting in the required structured field record lists.

Several methods may be applied to identify semi-structured record sets, such as MDR [8], DETPA [9], and RST [10]. MDR and DEPTA assume a fixed length of generalized nodes whereas RST relaxes this assumption by using a search structure called record segmentation tree which can dynamically generate subtree groups with different length. Moreover, RST provides a unified search based solution for region detection and record segmentation using a record segmentation tree structure. Our modified implementation of RST performs a top-down traversal detection in the DOM structure of a Web page.

After identifying semi-structured record sets, we make use of the partial tree alignment method [9] to conduct layout format driven alignment for the generation of structured field record lists. This approach aligns multiple tag trees of data records from the same record set by progressively growing a seed tree. The seed tree is chosen as the record tree with the largest number of data items because it is more likely for this tree to have a good alignment with data fields in other data records. Then the algorithm utilizes the seed tree as the core and aligns the remaining record trees with it one by one. We obtain the data fields from each record tree according to the alignment result and each record set is transformed into a structured field record list.

2.3 *Potential Entity Detection with Distant Supervision*

The aim of this component is to detect potential named entities for a particular record in a structured field record list. As mentioned above, we formulate it as a sequence classification problem, where each record is represented as a sequence of tokens and we aim at finding the label for each token. To achieve our goal, we make use of Conditional Random Field (CRF) [11] model. CRF is a discriminative undirected probabilistic graphical model, which enables us to include a large number of statistically correlated features. In particular we use linear-chain CRF, which considers conditional probability distribution $p(\mathbf{y}|\mathbf{x})$ of input sequence $\mathbf{x}$ and label sequence $\mathbf{y}$ as depicted in (1):

$$p_{\boldsymbol{\theta}}(\mathbf{y}|\mathbf{x}) = \frac{1}{Z_\theta(\mathbf{x})} \exp(\sum_k \theta_k F_k(\mathbf{x},\mathbf{y})), \tag{1}$$

where $Z_\theta(x) = \sum_y \exp(\sum_k \theta_k F_k(\mathbf{x},\mathbf{y}))$ is the partition function and $F_k(\mathbf{x},\mathbf{y}) = \sum_i f_k(\mathbf{x}, y_i, y_{i-1}, i)$ is the feature function. The most probable label sequence for a given input sequence $\mathbf{x}$ is

$$\mathbf{y} = \arg\max_{\mathbf{y}} p_{\boldsymbol{\theta}}(\mathbf{y}|\mathbf{x}) = \arg\max_{\mathbf{y}} \sum_k \theta_k F_k(\mathbf{y}, \mathbf{x}). \qquad (2)$$

As mentioned in the overview, we do not require training labels on the text data records. Instead, we leverage the existing large amount of labeled named entities from the external repository DBpedia. However, this labeled entities cannot be directly used as training data for our classification model. Instead, we incorporate this external clue via distant supervision by making use of Generalized Expectation (GE) constraints. GE constraints were first proposed in [12] to incorporate prior knowledge about the label distribution into semi-supervised learning, and were later used in document classification [13], information extraction [12], etc.

The idea of GE constraints is to make use of conditional probability distributions of labels given a feature. For example, we may specify the probability that the token ”George” labeled as PERSON should be larger than 80%. To capture this prior information, we introduce an auxiliary feature *f* as [[Entity Type=PERSON given Token=“George”]]. The corresponding affine constraint is $E_{p_\theta}[f(x,y)] \geq 0.8$. Learning with GE constraints will attempt to match this kind of label probability distribution for a particular feature by model expectation on the unlabeled data. The GE constraints objective function term is in the form of $\triangle(\hat{f}, E_{p_\theta}[f(x,y)])$, where $\triangle$ is a distance function; $\hat{f}$ is the target expectation; and p_θ is the model distribution. For the CRF model, we set the functions to be conditional probability distribution and set the distance function as KL-divergence between two distributions. By adding the constraint term to the standard CRF log-likelihood function, we can incorporate such kind of external prior knowledge during the training process.

In our framework, we add features that a given test segment matches an existing entity name in DBpedia, in the form of `B-DBpedia-X` and `I-DBpedia-X`, where `X` is the entity type associated with DBpedia. We set the feature target distribution that most text segments with these features are labeled as the corresponding entity type. We may have different expectations for different entity types. For example, we have high confidence that text segments appeared in the DBpedia species should be the SPECIES type, since species names are quite limited and specialized. Another example is that we allow the text segment with `DBpedia-Work` feature to be detected as WORK type at a relatively low target distribution. This is due to the nature of WORK type that entities in this type have more varieties. For example, *Jane Eyre* may be classified as WORK if we are talking the novel, or be classified as PERSON if we are talking the woman with this name. By making use of GE constraints to guide the model training, we are able to incorporate distant supervision from external repositories.

In the process of feature extraction, we also include some commonly used features employed in linear-chain sequence CRF models. These features include factors between each token and its corresponding label, neighboring tokens and labels, transition factors between labels and some word pattern features. The learning process will capture the importance of each feature.

2.4 *Collective Detection via Logical Inference*

As mentioned in the overview of our framework, we aim to make use of the inherent structure information to consider the consistency among potential named entities as well as header text in a field column. We investigate a model using first-order logic to conduct logical inference and make decision on the predicted entity type. The first-order logic aims at modeling the knowledge about the decision process that resembles how human beings conduct logical inference. Another characteristic of the decision making model is that we wish to allow a principled handling of uncertainty in the decision making knowledge as well as the inference process. To achieve our goal, we employ the Markov Logic Network (MLN) model [14] in this component.

MLN model combines the Markov network with first-order logic, enabling uncertain inference. A MLN, denoted as L, consists of a set of formulas with weights (F_i, w_i), where F_i is a formula expressed in first-order logic. Together with a set of constants $C = \{c_1, c_2, \ldots, c_{|C|}\}$, it defines a Markov network $M_{L,C}$ with binary-valued node. Given different sets of constants C, we get different Markov networks sharing the same structure and parameters. The generated Markov network is called a *ground Markov network*. The probability distribution over possible worlds x specified by the ground Markov network is given by

$$P(X = x) = \frac{1}{Z} \exp(\sum_i w_i n_i(x)) = \frac{1}{Z} \prod_i \phi_i(x_i)^{n_i(x)}, \tag{3}$$

where $n_i(x)$ is the number of true groundings of F_i in x. Given a ground Markov network, we can query the probability of whether a given ground atom is true. This inference procedure can be performed by MCMC over the minimal set of the ground network required to answer the query.

In our framework, we employ MLN to capture the following knowledge in the collective inference component:

- Potential named entities under the same field column tend to share the same entity type. This observation is derived from the inherent structure of record lists.
- If a given field column contains multiple potential entities, they likely share the same entity type. This is generally true due to the nature of the field such as the "Keynote speakers" column in Fig. 2.
- Potential named entities in the same field column should be consistent with the header text. For example, if header text is "Keynote speakers", the named entities under the column likely belong to the entity type PERSON.

Header text provides extremely useful clues for entity detection. To effectively make use of header information, we develop a method to incorporate header text with uncertainty handling by using the hypernym tree of an ontology such as WordNet [15]. In the beginning, we manually associate a set of ontology concepts for each entity type $c \in \mathcal{C}$, denoted as OC_c according to the intended meaning of the entity types for the application. For example, OC_{WORK} contains the concepts "painting, picture (3876519)" and "album, record album (6591815)", where each concept is denoted by the synonym set with the concept ID in the parenthesis. Given an input header text in the form of noun phrase, we preprocess the header text with noun phrase chunker and identify the core term, denoted as ct. If the core term is in the plural form, its singular form is returned. For example, the term "speaker" in "Keynote speakers" is identified as the core term. Then we lookup the core term in the hypernym tree of WordNet to obtain the concepts that contain the core term, detored as OC_{ct}. Let $OC_{ct,c}$ denote the concepts in OC_{ct} that are in the hyponym paths of the concepts in OC_c. Let $\mathcal{C}' = \mathcal{C} \cup \{\text{NON-ENTITY}\}$, and $OC_{ct,\text{NON-ENTITY}}$ denote the concepts in OC_{ct} that are not in the hyponym paths of any concept in OC_c. The probability that the core term ct is associated with an entity type c is calculated as:

$$P(c|ct) = \frac{OC_{ct,c}}{\sum_{c' \in \mathcal{C}'} OC_{ct,c'}}. \tag{4}$$

To combine different clues, we define the predicates as shown in Table 1. The variable `entity` represents the detected potential named entities; `column` represents the field column; `type` represents the entity

Table 1. List of MLN predicates

Predicate	Meaning
EntityInColumn(entity, column)	column information
ColumnHeaderSimilarToType (column, type)	header information
ColumnDominantType(column, type)	column dominant entity type
EntityInitialType(entity, type)	initial type given by detection phrase
EntityFinalType(entity, type)	final type after logical inference

types. We design the following logical formulas, namely, from LF1 to LF4.

The formula LF1 expresses an observation corresponding to a field column:

$$\textsc{EntityInColumn}(\textsc{e,c}) \wedge \textsc{EntityInitialType}(\textsc{e,t}) \Rightarrow \textsc{ColumnDominantType}(\textsc{c,t}) \qquad \text{(LF1)}$$

The more detected named entities from a single column that share the same entity type, the more likely that the field column contains that type of entities. A field column may contain multiple types of entities, each detected entity will contribute to the column global entity type. Note that the "+" symbol beside the variable T means that we will expand this formula with each possible groundings of T.

The formula LF2 incorporates the column header information for a given column:

$$\textsc{ColumnHeaderSimilarToType}(\textsc{c,t}) \Rightarrow \textsc{ColumnDominantType}(\textsc{c,t}) \qquad \text{(LF2)}$$

If the associate probability of the header text in the column C with an entity type T expressed in Equation (4) exceeds a threshold, then we add the corresponding positive evidence predicate ColumnHeaderSimilarToType(C,T). Note that header text may indicate multiple potential entity types. For example header text "Member" may contain list of organizations, or list of person names. Together with the formula LF1, we can infer the probability of global entity type for a field column.

The formula LF3 indicates that the final entity type for a potential named entity E tend to be consistent with the original one:

$$\textsc{EntityInitialType}(\textsc{e,t}) \Rightarrow \textsc{EntityFinalType}(\textsc{e,t}) \qquad \text{(LF3)}$$

We observe that our sequence classification model can detect most of the named entities correctly, thus we give this formula a relatively high weight.

Besides the original type given during the detection phrase, the final entity type also depends on the column C where the entity E is located as shown in LF4:

$$\text{EntityInColumn(e,c)} \wedge \text{ColumnDominantType(c,t)} \Rightarrow \text{EntityFinalType(e,t)} \quad \text{(LF4)}$$

Field labels tend to be consistent with the column global entity type. The influence of column global entity type will increase as we have higher confidence on column entity type.

We can handle the situation that a column may have multiple global named entities. In this case, each field contains multiple named entities with different types.

3 Experiment

3.1 *Experiment Setup*

For the semi-structure record sets, we harvested from Web as described in Section 2.2. For the tabular record sets, we collected from a subset of the table corpus as mentioned in [16]. As a result, we collected 3,372 semi-structured and tabular record sets in total. Note that all these record sets do not have training labels. The number of records in a record set ranges from 2 to 296, with average 30. For the purpose of evaluation, we recruited annotators to find the ground truth named entities and provide labels on a subset of our full dataset. The number of record sets in this evaluation set is 650 composed of 16,755 true named entities.

We focused on the detection of five types of named entity: Organization, Person, Place, Work, Species. The meaning of these five types is exactly the same as in DBpedia. For example, Work includes artistic creations such as films, albums or songs. The remaining entity types are self-explanatory. We used DBpedia 3.8 published in August 2012 and indexed all the entity names using Apache Lucene for fast lookup when extracting CRF features.

We also implemented a comparison model known as *Repository Supervised Model*. This model checks each text segment against DBpedia and finds the corresponding entity type if exists. If a text segment corresponds to multiple named entities of different types in DBpedia, we randomly selected one.

Besides our full model, we also investigate a model known as *Our Model Without Collective Inference*. This model is essential our proposed model, but omitting the collective inference part. By comparing our proposed model with this one, we can investigate the benefit of the collective inference component.

We implemented the sequence classification model based on the open source MALLET [17] package, which provides implementation for linear-chain CRF with GE constraints. The collective logical inference is implemented based on the Alchemy[3] package, which provides functions for MLN inference. We manually assign weights to the formulas based on our prior knowledge. Specifically, we set w_1 as 1.0, w_2 as 5.0, w_3 as 2.0, and w_4 as 1.0. Our experiments show that the parameters are not sensitive to the final performance much.

3.2 *Evaluation result*

We use standard evaluation metrics, namely, precision P, recall R, and their harmonic mean F1 where $F1 = 2 \times P \times R/(P + R)$. We followed CoNLL-2003 evaluation procedure which only counts the exact match for entity names. Table 2 shows the performance of our experiment.

From the evaluation result, it is clear that our proposed framework outperforms the Repository Supervised model significantly by over 20% relative F1 score improvement. The average recall for the Repository Supervised Model is only around 40%, meaning that more than half of the named entities in the evaluation set are not present in DBpedia. Our proposed framework successfully detects many previously unseen named entities with high precision.

Compared to the Repository Supervised model, our model without collective inference still improves the performance by about 10%. This result demonstrates the effectiveness of the sequence classification model, which can capture large amount of features such as word capitalization, neighborhood labels, and boundary tokens across the record. Even though we do not use any labeled records as training data, the distant supervision with existing repository named entities still leads to good performance.

[3] Available at http://alchemy.cs.washington.edu

Table 2. Experimental result

Model	Measure	Organization	Person	Place	Species	Work	Overall
Repository Supervised Model	Precision	61.63%	78.33%	26.31%	93.05%	54.34%	60.44%
	Recall	50.06%	42.05%	11.10%	32.25%	44.55%	38.56%
	F1-score	55.24%	54.73%	15.62%	47.90%	48.96%	47.08%
Our Model w/o Collective Inference	Precision	75.95%	64.77%	44.81%	89.43%	68.32%	66.31%
	Recall	70.60%	56.90%	17.21%	100.00%	48.63%	48.70%
	F1-score	73.18%	60.58%	24.86%	94.42%	56.81%	56.16%
Our Full Model	Precision	69.54%	72.63%	81.18%	100.00%	64.87%	70.46%
	Recall	83.17%	75.99%	44.64%	100.00%	86.40%	74.79%
	F1-score	75.74%	74.27%	57.60%	100.00%	86.40%	72.56%

With the collective inference component, our full model further improves the performance. By taking advantage of the inherent structure of record set, we can discover more named entities with higher precision.

4 Related Work

Some methods have been proposed to detect entities from Web pages. For example, Limaye et al. developed a system that can find entities and relationships [16]. It mainly recognizes terms in the Web content that are some known entities found in a database, known as a catalog. The main characteristic of their method is to allow approximate matching between the terms in the Web text and the entity in the catalog. Kulkarni et al. proposed a method for matching spots on Web pages to Wikipedia entities [18]. However, all these methods dealing with Web texts assume that all potential entities detected are known entities. In contrast, our proposed framework is able to detect entities not already seen before.

Recently, researchers explore another valuable information resource, namely search log, in order to conduct entity extraction or attribute acquisition [19–22]. In [19], a seed-based framework was proposed to allow weakly supervised extraction of named entities from Web search queries by calculating the similarity score between the search-signature vector of a candidate instance and the reference search-signature vector of a seed class. In [21], Guo et al. attempted to use a topic model to identify named entities in queries, and they showed that around 70% of the real search queries contain named entities. The methods in the above works are not applicable for the task we tackle in this paper due to data characteristics.

Currently, the state-of-the-art method for NER from free text is based on Conditional Random Fields [2, 23]. This approach is already applied in the entity detection flourishing short tweets under the combination with other models [24, 25]. However, these works are not suitable for our text content due to the nature of text data records. Moreover, we do not have manual labels on the text data records. In addition, the inter-dependency among the records in the same record set cannot be taken into account in traditional NER methods.

Distant supervision has been employed in various tasks such as relation extraction [26, 27], sentiment analysis [28, 29], and entity extraction from advertisements or tweets [30, 31]. As far as we know, our work is the first one that applies distant supervision on entity extraction from semi-structured data records using the generalized expectation model.

5 Conclusions and Future Work

We have proposed a new framework for detecting named entities from semi-structured web data including semi-structured and tabular record sets. We transform them into a unified representation, and then use a primarily unsupervised CRF model trained with GE constraints. We also propose a collective logical inference method that enables us to incorporate the underlying structure and header text information in record lists. We demonstrate the effectiveness of our framework through extensive experiments.

We intend to develop a more efficient training algorithm. Currently CRF training with GE constraints can only handle local features. Therefore we need to use MLN to incorporate global constraints. We will investigate an integrated way to handle such capability in a unified manner.

ACKNOWLEDGMENTS The work is supported by a grant from the Research Grant Council of the Hong Kong Special Administrative Region, China (Project Code: CUHK413510).

References

1. McCallum, A., Li, W.: Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons. In: Proceedings of the seventh conference on Natural language learning at HLT-NAACL. (2003)
2. Sarawagi, S., Cohen, W.W.: Semi-markov conditional random fields for information extraction. In: NIPS. (2004) 1185–1192
3. McNamee, P., Snow, R., Schone, P., Mayfield, J.: Learning named entity hyponyms for question answering. In: Proc. of the Third International Joint Conference on Natural Language Processing. (2008) 799–804
4. Pasca, M.: Weakly-supervised discovery of named entities using web search queries. In: Proc. of CIKM. (2007)
5. Suzuki, J., Isozaki, H.: Semi-supervised sequential labeling and segmentation using giga-word scale unlabeled data. In: Proc. of ACL-08: HLT
6. Cafarella, M.J., Halevy, A., Wang, D.Z., Wu, E., Zhang, Y.: Webtables: exploring the power of tables on the web. Proc. VLDB Endow. **1**(1) (August 2008) 538–549
7. Ramshaw, L.A., Marcus, M.P.: Text chunking using transformation-based learning. CoRR (1995)
8. Liu, B., Grossman, R., Zhai, Y.: Mining data records in web pages. In: Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining. KDD (2003) 601–606
9. Zhai, Y., Liu, B.: Structured data extraction from the web based on partial tree alignment. IEEE Trans. on Knowl. and Data Eng. **18**(12) (December 2006)
10. Bing, L., Lam, W., Gu, Y.: Towards a unified solution: data record region detection and segmentation. In: Proceedings of the 20th ACM international conference on Information and knowledge management. CIKM '11 (2011)
11. Lafferty, J.D., McCallum, A., Pereira, F.C.N.: Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In: Proceedings of the Eighteenth International Conference on Machine Learning. (2001) 282–289
12. Mann, G.S., McCallum, A.: Simple, robust, scalable semi-supervised learning via expectation regularization. In: Proceedings of the 24th international conference on Machine learning. ICML '07 (2007) 593–600
13. Druck, G., Mann, G., McCallum, A.: Learning from labeled features using generalized expectation criteria. In: Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval. (2008)
14. Richardson, M., Domingos, P.: Markov logic networks. Mach. Learn. **62**(1-2) (February 2006) 107–136
15. Miller, G.A.: WordNet: a lexical database for english. Commun. ACM **38**(11) (November 1995) 39–41

16. Limaye, G., Sarawagi, S., Chakrabarti, S.: Annotating and searching web tables using entities, types and relationships. Proc. VLDB Endow. **3**(1-2) (2010)
17. McCallum, A.K.: MALLET: A machine learning for language toolkit. http://mallet.cs.umass.edu (2002)
18. Kulkarni, S., Singh, A., Ramakrishnan, G., Chakrabarti, S.: Collective annotation of wikipedia entities in web text. In: Proc. of the Int. Conf. on Knowledge Discovery and Data Mining. (2009) 457–465
19. Paşca, M.: Weakly-supervised discovery of named entities using web search queries. In: Proceedings of the sixteenth ACM conference on Conference on information and knowledge management. CIKM '07 (2007) 683–690
20. Paşca, M., Durme, B.V.: Weakly-supervised acquisition of open-domain classes and class attributes from web documents and query logs. In: ACL. (2008) 19–27
21. Guo, J., Xu, G., Cheng, X., Li, H.: Named entity recognition in query. In: Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval. SIGIR '09 (2009) 267–274
22. Jain, A., Pennacchiotti, M.: Open entity extraction from web search query logs. In: Proceedings of the 23rd International Conference on Computational Linguistics. COLING '10 (2010) 510–518
23. Krishnan, V., Manning, C.D.: An effective two-stage model for exploiting non-local dependencies in named entity recognition. In: Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics. ACL-44 (2006)
24. Liu, X., Zhang, S., Wei, F., Zhou, M.: Recognizing named entities in tweets. In: Proceedings of the 49th Annual Meeting of the ACL: Human Language Technologies. HLT '11 (2011)
25. Ritter, A., Clark, S., Etzioni, M., Etzioni, O.: Named Entity Recognition in Tweets: An Experimental Study. In: 2011 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics (2011)
26. Mintz, M., Bills, S., Snow, R., Jurafsky, D.: Distant supervision for relation extraction without labeled data. In: Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP. ACL '09 (2009) 1003–1011
27. Surdeanu, M., McClosky, D., Tibshirani, J., Bauer, J., Chang, A.X., Spitkovsky, V.I., Manning, C.D.: A simple distant supervision approach for the tac-kbp slot filling task. In: Proceedings of the TAC-KBP 2010 Workshop. (2010)
28. Purver, M., Battersby, S.: Experimenting with distant supervision for emotion classification. In: Proceedings of the 13th Conference of the EACL. (2012)

29. Marchetti-Bowick, M., Chambers, N.: Learning for microblogs with distant supervision: Political forecasting with twitter. In: EACL. (2012) 603–612
30. Singh, S., Hillard, D., Leggetter, C.: Minimally-supervised extraction of entities from text advertisements. In: Human Language Technologies: The 2010 Annual Conference of the NAACL. HLT '10 (2010) 73–81
31. Li, C., Weng, J., He, Q., Yao, Y., Datta, A., Sun, A., Lee, B.S.: Twiner: named entity recognition in targeted twitter stream. In: Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval. SIGIR '12 (2012) 721–730

CHUNLIANG LU
THE CHINESE UNIVERSITY OF HONG KONG,
HONG KONG
E-MAIL: <CLLU@SE.CUHK.EDU.HK>

LIDONG BING
THE CHINESE UNIVERSITY OF HONG KONG,
HONG KONG
E-MAIL: <LDBING@SE.CUHK.EDU.HK>

WAI LAM
THE CHINESE UNIVERSITY OF HONG KONG,
HONG KONG
E-MAIL: <WLAM@SE.CUHK.EDU.HK>

KI CHAN
HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY,
HONG KONG
E-MAIL: <KCCECIA@CSE.UST.HK>

YUAN GU
THE CHINESE UNIVERSITY OF HONG KONG,
HONG KONG
E-MAIL: <YUANGU@SE.CUHK.EDU.HK>

Natural Language Generation and Grammar Checking

IJCLA VOL. 4, NO. 2, JUL-DEC 2013, PP. 153–167
RECEIVED 07/12/12 ACCEPTED 04/03/13 FINAL 05/03/13

Japanese Sentence Order Estimation using Supervised Machine Learning with Rich Linguistic Clues

YUYA HAYASHI, MASAKI MURATA, LIANGLIANG FAN, AND MASATO TOKUHISA

Tottori University, Japan

ABSTRACT

Estimation of sentence order (sometimes referred to as sentence ordering) is one of the problems that arise in sentence generation and sentence correction. When generating a text that consists of multiple sentences, it is necessary to arrange the sentences in an appropriate order so that the text can be understood easily. In this study, we proposed a new method using supervised machine learning with rich linguistic clues for Japanese sentence order estimation. As one of rich linguistic clues we used concepts on old information and new information. In Japanese, we can detect phrases containing old/new information by using Japanese topic-marking postpositional particles. In the experiments of sentence order estimation, the accuracies of our proposed method (0.72 to 0.77) were higher than those of the probabilistic method based on an existing method (0.58 to 0.61). We examined features using experiments and clarified which feature was important for sentence order estimation. We found that the feature using concepts on old information and new information was the most important.

KEYWORDS: *Sentence order estimation, supervised machine learning, linguistic clue, old / new information*

1 INTRODUCTION

Estimation of sentence order (sometimes referred to as sentence ordering) is one of the problems that arise on sentence generation and sentence correction [1–6]. When generating a text that consists of multiple sentences, it is necessary to arrange the sentences in an appropriate order so that the text can be understood easily.

Most of the studies on sentence order estimation were for multi document summarization, and they used the information obtained from the original sentences before summarizing for estimating sentence order [7–21]. If we can estimate sentence order without the original sentences before summarizing, the technique of estimating sentence order can be utilized for a lot of applications (e.g., sentence correction). For example, a text where the order of sentences is not good can be modified into a text where the order of sentences is good. Furthermore, the grammatical knowledge on sentence order will be able to be obtained through the study on sentence order without the original sentences. For example, when we find that a feature using a linguistic clue is important in the study on sentence order estimation, we can acquire the grammatical knowledge that the linguistic clue is important in sentence order estimation. Therefore, in this study, we handle the sentence order estimation that does not use the information on the original sentences before summarizing. In a study about sentence order estimation without using the original sentences before summarizing, Lapata proposed a probabilistic model [22]. However, supervised machine learning has not been used for that estimation. Therefore, in this study, we use supervised machine learning for sentence order estimation without using the original sentences before summarizing. In this study, we use the support vector machine (SVM) as the supervised machine learning [23].

We propose a method of sentence order estimation using numerous linguistic clues besides supervised machine learning. It is difficult for a probabilistic model to use a lot of information. In contrast, when using supervised learning, we can very easily use a lot of information by preparing many features. Because our proposed method uses a lot of information, it can be expected that our proposed method outperforms the existing method based on a probabilistic model.

In this paper, we use a simple task for sentence order estimation. We consider that the phenomenon across multiple paragraphs is complicated. We handle the problem where we judge which sentence we should write first among two sentences in a paragraph using the information in the paragraph.[1]

In this study, we handle sentence order estimation in Japanese.

We present the main points of this study as follows:

1. Our study has originality, and used supervised machine learning for sentence order estimation with rich linguistic clues for the first time. As one of rich linguistic clues we used features based on concepts of old information and new information.
2. We confirmed that the accuracy rates of our proposed method using supervised machine learning (0.72 to 0.77) was higher than those of the existing methods based on a probabilistic model (0.58 to 0.61). Our proposed method has a high usability because the performance accuracy was high.
3. Our proposed method using supervised learning can use a lot of features (information) easily. It is expected that our method improves the performance by using more features.
4. In our proposed method using supervised learning, we can find important features (information) in sentence order estimation by examining features. When we examined features in our experiments, we found that the feature based on the concept of old/new information. The feature checked the number of common content words between the subject in the second sentence and the part after the subject in the first sentence is the most important in sentence order estimation.

2 RELATED STUDIES

In a study [22] that is similar to ours, Lapata proposed a probabilistic model for sentence order estimation that did not use the original sentences before summarizing. Lapata calculated the probabilities of sentence occurrences using the probabilities of word occurrences, and estimated sentence orders by the probabilities of sentence occurrences.

Most of the studies on sentence order estimation are for multi document summarization, and they use the information obtained from the original sentences before summarizing for estimating sentence order [8, 9, 13, 19, 21]. Bollegala et al. performed sentence order estimation against the sentences that were extracted from multiple documents. They used

[1] An estimate of the order of all the sentences in a full text would be handled by combining estimated orders in pairs of two sentences.

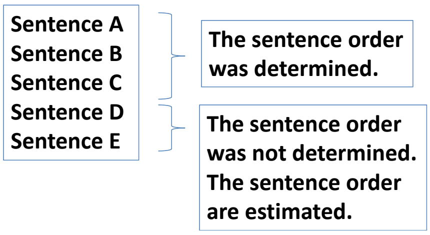

Sentence A
Sentence B
Sentence C
Sentence D
Sentence E
The sentence order was determined.
The sentence order was not determined. The sentence order are estimated.

Fig. 1. The model of the task

original documents before summarization for sentence order estimation. They focused on how the sentences, whose order would be estimated, were located in original documents before summarization. In addition, they used chronological information and topical-closeness. They used supervised machine learning for combining these kinds of information. However, they did not use linguistic clues such as POSs (parts of speech) of words and a concept on linguistic old/new information (related to subjects and Japanese postpositional particles) as features for machine learning.

Uchimoto et al. studied word order using supervised machine learning [24]. They used linguistic clues such as words and parts of speech as features for machine learning. They used machine learning for word order estimation. In contrast, we used machine learning for sentence order estimation. They estimated word order using word dependency information. Correct word orders are in corpora. Therefore, the training data on word order can be constructed from corpora automatically. In a similar way, the training data on sentence order can be constructed from corpora automatically. In our study, we use the training data that are constructed from corpora automatically.

3 The task and the proposed method

3.1 *The task*

The task in this study is as follows: a paragraph is input, the order of the first several sentences in the paragraph is determined, the order of the remaining sentences in the paragraph is not determined, and the estimation of the order of two sentences among the remaining sentences is the task. The information that can be used for estimation is the two sentences

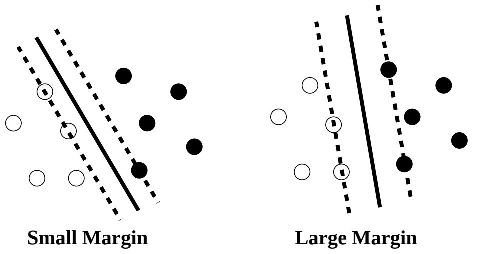


Fig. 2. Maximizing the margin

whose order will be estimated, and the sentences before one of the two sentences appears in the paragraph (see Figure 1).

3.2 *Our proposed method*

We assume that we need to estimate the order of two sentences, A and B. These sentences are input in the system and our method judges whether the order of "A-B" is correct by using supervised learning. In this study, we use SVM as machine learning. We use a quadratic polynomial kernel as a kernel function.

The training data is composed as follows: two sentences are extracted from a text that is used for training. From the two sentences, a sequence of the two sentences with the same order as in an original text, and a sequence of the two sentences with the reverse order are made. The two sentences with the same order are used as a positive example, and the two sentences with the reverse order are used as a negative example.

3.3 *Support vector machine method*

In this method, data consisting of two categories is classified by dividing space with a hyperplane. When the margin between examples which belong to one category and examples which belong to the other category in the training data is larger (see Figure 2[2]), the probability of incorrectly choosing categories in open data is thought to be smaller. The hyperplane

[2] In the figure, the white circles and black circles indicate examples which belong to one category and examples which belong to the other category, respectively. The solid line indicates the hyperplane dividing space, and the broken lines indicate planes at the boundaries of the margin regions.

maximizing the margin is determined, and classification is done by using this hyperplane. Although the basics of the method are as described above, for extended versions of the method, in general, the inner region of the margin in the training data can include a small number of examples, and the linearity of the hyperplane is changed to non-linearity by using kernel functions. Classification in the extended methods is equivalent to classification using the following discernment function, and the two categories can be classified on the basis of whether the output value of the function is positive or negative [23, 25]:

$$\begin{aligned} f(\mathbf{x}) &= sgn\left(\sum_{i=1}^{l} \alpha_i y_i K(\mathbf{x}_i, \mathbf{x}) + b\right) \\ b &= -\frac{max_{i,y_i=-1} b_i + min_{i,y_i=1} b_i}{2} \\ b_i &= \sum_{j=1}^{l} \alpha_j y_j K(\mathbf{x}_j, \mathbf{x}_i), \end{aligned} \tag{1}$$

where $\mathbf{x}$ is the context (a set of features) of an input example; $\mathbf{x}_i$ and $y_i (i = 1, ..., l, y_i \in \{1, -1\})$ indicate the context of the training data and its category, respectively; and the function sgn is defined as

$$\begin{aligned} sgn(x) &= 1 \ \ (x \geq 0), \\ & \quad -1 \ (otherwise). \end{aligned} \tag{2}$$

Each $\alpha_i (i = 1, 2...)$ is fixed when the value of $L(\alpha)$ in Equation (3) is maximum under the conditions of Equations (4) and (5).

$$L(\alpha) = \sum_{i=1}^{l} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{l} \alpha_i \alpha_j y_i y_j K(\mathbf{x_i}, \mathbf{x_j}) \tag{3}$$

$$0 \leq \alpha_i \leq C \ (i = 1, ..., l) \tag{4}$$

$$\sum_{i=1}^{l} \alpha_i y_i = 0 \tag{5}$$

Although the function K is called a kernel function and various types of kernel functions can be used, this paper uses a polynomial function as follows:

$$K(\mathbf{x}, \mathbf{y}) = (\mathbf{x} \cdot \mathbf{y} + 1)^d, \tag{6}$$

where C and d are constants set by experimentation. In this paper, C and d are fixed as 1 and 2 for all experiments, respectively.[3] A set of $\mathbf{x}_i$ that satisfies $\alpha_i > 0$ is called a support vector, and the portion used to perform the sum in Equation (1) is calculated by only using examples that are support vectors.

We used the software TinySVM [25] developed by Kudoh as the support vector machine.

3.4 *Features used in our proposed method*

In this section, we explain features (information used in classification), which are required to use machine learning methods.

Features used in this study are shown in Table 1. Each feature has additional information of whether it appears in the first or second sentence. The first and the second sentence that are input are indicated with A and B, respectively.

Concretely speaking, we used a topic instead of a subject for F9. The part before a Japanese postpositional particle *wa* indicates a topic. We used the number of the common content words between the part before *wa* in the second sentence B and the part after *wa* in the first sentence for F9.

F9 is a feature based on a concept of old/new information. Because the part before a Japanese postpositional particle *wa* indicates a topic, it is likely to contain old information and the part after a Japanese postpositional particle *wa* is likely to contain new information. A Japanese postpositional particle *wa* in “Noun X *wa*” is similar to an English prepositional phrase “in terms of” in “in terms of Noun X” and indicates that “Noun X” is a topic. In correct sentence order, words in a part containing old information of the second sentence are likely to appear in a part containing new information of the first sentence. Based the above idea, we used F9.

[3] We confirmed that $d = 2$ produced good performance in preliminary experiments.

Table 1. Feature

ID	Definition
F1	The words and their parts of speech (POS) in the sentence A (or B).
F2	The POS of the words in the sentence A (or B).
F3	Whether the subject is omitted in the sentence A (or B).
F4	Whether a nominal is at the end of the sentence A (or B).
F5	The words and their POS in the subject of the sentence A (or B).
F6	The words and their POS in the part after the subject in the sentence A (or B).
F7	The pair of the postpositional particles in the two sentences A and B.
F8	The number of common content words between the two sentences A and B.
F9	The number of common content words between the subject in the second sentence B and the part after the subject in the first sentence A.
F10	The words and their POS in all the sentences before the two sentences A and B in the paragraph.
F11	Whether a nominal is at the end of the sentence just before the two sentences A and B in the paragraph.
F12	Whether the subject is omitted in the sentence just before the two sentences A and B in the paragraph.
F13	The number of the common content words between the sentence just before the two sentences A and B in the paragraph and the sentence A (or B).

4 Probabilistic Method (Compared Method)

We compare our proposed method based on machine learning with the probabilistic method. Here, the probabilistic method is based on Lapata's method using probabilistic models [22].

The detail of the probabilistic method is as follows: words that appear in two adjacent sentences are extracted from a text that is used for calculating probabilities. All the pairs of a word W_A in the first sentence, and a word W_B in the second sentence are made. Then the occurrence probability that when a word W_A appears in a first sentence, a word W_B appears in a second sentence is calculated for each word pair. The occurrence probability (that we call sentence occurrence probability) that the second sentence appears when the first sentence is given is calculated by multiplying the probabilities of all the word pairs. In this study, to estimate the order for two sentences A and B, a pair $Pair_{AB}$ with the original order (A-B) and a pair $Pair_{BA}$ with the reverse order (B-A) are generated. When the sentence occurrence probability of $Pair_{AB}$ is

Table 2. The number of pairs of two sentences

	CASE1	CASE2	CASE3
Training data	33902	64290	130316
Test data	40386	82966	170376

larger than that of $Pair_{BA}$, the method judges that the order of $Pair_{AB}$ is correct. Otherwise, it judges that the order of $Pair_{BA}$ is correct.

$a_{\langle i,1\rangle}$, .., $a_{\langle i,n\rangle}$ indicate to the words that appear in a sentence S_i. The probability that $a_{\langle i,j\rangle}$ and $a_{\langle i-1,k\rangle}$ appear in the two adjacent sentences are expressed in the following equation: equation:

$$P(a_{\langle i,j\rangle}|a_{\langle i-1,k\rangle}) = \frac{f(a_{\langle i,j\rangle}, a_{\langle i-1,k\rangle})}{\sum_{a\langle i,j\rangle} f(a_{\langle i,j\rangle}, a_{\langle i-1,k\rangle})} \tag{7}$$

$f(a_{\langle i,j\rangle}, a_{\langle i-1,k\rangle})$ is the frequency that a word $a_{\langle i,j\rangle}$ appears in the sentence just after the sentence having a word $a_{\langle i-1,k\rangle}$.

When there is a sentence C just before sentences whose order will be estimated, the sentence occurrence probability of $Pair_{AB}$ is multiplied by the sentence occurrence probability of sentence A appearing just after sentence C.

5 Experiment

5.1 *Experimental condition*

We used Mainichi newspaper articles (May, 1991) for the machine learning of the training data. We used Mainichi newspaper articles (November, 1995) for the test data. We used Mainichi newspaper articles (1995) for the text that is used for calculating probabilities in the probabilistic method.

We used the following three kinds of cases for pairs of two sentences used in the experiments: CASE 1: We made pairs of two sentences by using only the first two sentences in a paragraph. CASE 2: We made pairs of two sentences by using all the adjacent two sentences in a paragraph. CASE 3: We made pairs of two sentences by using all the two sentence combinations in a paragraph.

The number of pairs of two sentences used in the training and test data are shown in Table 2.

Table 3. Accuracy

Machine learning (ML)			Probabilistic method (PM)		
CASE1	CASE2	CASE3	CASE1	CASE2	CASE3
0.7677	0.7246	0.7250	0.6059	0.5835	0.5775

Table 4. Comparison with accuracies of human subjects

	Subjects						ML	PM
	A	B	C	D	E	Ave.		
CASE1	0.75	0.70	0.75	0.95	0.95	0.82	0.79	0.65
CASE2	0.80	0.80	0.85	1.00	0.90	0.87	0.67	0.64
CASE3	0.65	0.75	0.85	0.65	0.70	0.72	0.71	0.56

5.2 *Experimental results*

The accuracies of our proposed method and the probabilistic method are shown in Table 3. As shown in Table 3, the accuracies of our proposed method (0.72 to 0.77) were higher than those of the probabilistic method (0.58 to 0.61).

5.3 *Comparison with accuracies of manual sentence order estimation*

We randomly extracted 100 pairs (each pair consists of two sentences) from Mainichi newspaper articles (November, 1995), and each of the five subjects estimated the order of 20 pairs among the 100 pairs for each of the CASEs 1 to 3. Our proposed method (ML) and the probabilistic method (PM) estimated the orders of 100 pairs. In CASE 2 and CASE 3, because the information on sentences was used in the supervised learning and the probabilistic methods, the sentences before two sentences whose orders will be estimated are shown to subjects.

Accuracies of subjects, ML, and PM are shown in Table 4. “A” to “E” in the table indicate the five subjects. “Average” indicates the average of accuracies of the five subjects.

When we compared the average accuracies of the subjects, and the accuracy of our proposed method (ML) in Table 4, we found that our proposed method could obtain accuracies that were very similar to the average accuracies of the subjects in CASEs 1 and 3.

Table 5. Accuracies of eliminating a feature

Eliminated feature	Accuracy	Difference
F1	0.7211	-0.0039
F2	0.7226	-0.0024
F3	0.7251	+0.0001
F4	0.7251	+0.0001
F5	0.7212	-0.0038
F6	0.7223	-0.0027
F7	0.7243	-0.0007
F8	0.7201	-0.0049
F9	0.6587	-0.0663
F10	0.7172	-0.0078
F11	0.7240	-0.0010
F12	0.7241	-0.0009
F13	0.7241	-0.0009

5.4 *Analysis of features*

Among the features used in this study, we examined which feature was useful for sentence order estimation. We compared accuracies of eliminating a feature and the accuracy of using all the features in CASE 3. Table 5 shows the accuracies of eliminating a feature. It also shows the result of subtracting the accuracy using all the features from the accuracies after eliminating a feature.

From Table 5, we found that the accuracy went down heavily without feature F9. We found that feature F9 was particularly important in sentence order estimation. An example that the estimation succeeds when using F9 and the estimation fails when not using F9 is shown as follows:

Sentence 1:
kotani-san-niwa hotondo chichi-no kioku-ga nai.
(Kotani) (almost) (father) (recollection) (no)
(Kotani has very few recollection of his father.)

Sentence 2:
chichi-ga byoushi-shita-no wa gosai-no toki-datta.
(father) (died of a disease) (five years old) (was when)
(The time that his father died of a disease was when he was five years old.)

The correct order is "Sentence 1 to Sentence 2." No use of F9 estimated that the order was "Sentence 2 to Sentence 1." F9 is the feature that checks the number of common content words between the subject in the second sentence and the part after the subject in the first sentence. Because "*chichi*" (father) appeared at the subject in the second sentence and the part after the subject in the first sentence, the use of F9 could estimate the correct order of the above example.

F9 is based on concepts of old/new information. In our method, we obtained good results on sentence order estimation by using the feature (F9) based on concepts of old/new information. A Japanese word *wa* in the phrase *byoushi-shita-no wa* (died of a disease) is a postpositional particle indicating a topic. A phrase *chichi-ga byoushi-shita-no wa* (father, died of a disease) is a topic part indicated by *wa* and corresponds to old information. Old information must appear in a previous part. "*chichi*" (father) appearing in a phrase corresponding to old information of Sentence 2 appears in Sentence 1. Therefore, the sentence order of "Sentence 1 to Sentence 2" is good. Our method using F9 can handle the concepts of old/new information and accurately judge the sentence order of the above example.

6 Conclusion

In this study, we proposed a new method of using supervised machine learning for sentence order estimation. In the experiments of sentence order estimation, the accuracies of our proposed method (0.72 to 0.77) were higher than those of the probabilistic method based on an existing method (0.58 to 0.61). When examining features, we found that the feature that checked the number of common content words between the subject in the second sentence, and the part after the subject in the first sentence was the most important in sentence order estimation. The feature is based on concepts of old/new information.

In the future, we would like to improve the performance of our method by using more features for machine learning. Furthermore, we would like to detect more useful features in addition to the feature based on concepts of old/new information. Useful detected features can be used as grammatical knowledge for sentence generation.

In this study, we handled the information within a paragraph. However, we should use information outside a paragraph when we handle orders of sentences in a full text. We should also consider sentence order estimation of two sentences across multiple paragraphs and estimation of the order of paragraphs. In the future, we would like to handle such things.

ACKNOWLEDGMENTS This work was supported by JSPS KAKENHI Grant Number 2350 0178.

REFERENCES

1. Duboue, P.A., McKeown, K.R.: Content planner construction via evolutionary algorithms and a corpus-based fitness function. In: Proceedings of the second International Natural Language Generation Conference (INLG '02). (2002) 89–96
2. Karamanis, N., Manurung, H.M.: Stochastic text structuring using the principle of continuity. In: Proceedings of the second International Natural Language Generation Conference (INLG '02). (2002) 81–88
3. Mann, W.C., Thompson, S.A.: Rhetorical structure theory: Toward a functional theory of text organization. Text **8** (1988) 243–281
4. Marcu, D.: From local to global coherence: A bottom-up approach to text planning. In: Proceedings of the 14th National Conference on Artificial Intelligence. (1997) 629–635
5. Marcu, D.: The rhetorical parsing of unrestricted texts: A surface-based approach. Computational Linguistics **26** (2000) 395–448
6. Murata, M., Isahara, H.: Automatic detection of mis-spelled japanese expressions using a new method for automatic extraction of negative examples based on positive examples. IEICE Transactions on Information and Systems **E85–D** (2002) 1416–1424
7. Barzilay, R., Elhadad, N., McKeown, K.R.: Inferring strategies for sentence ordering in multidocument news summarization. Journal of Artificial Intelligence Research **17** (2002) 35–55
8. Barzilay, R., Lee, L.: Catching the drift: Probabilistic content models, with applications to generation and summarization. In: Proceedings of HLT-NAACL 2004. (2004) 113–120
9. Bollegala, D., Okazaki, N., Ishizuka, M.: A bottom-up approach to sentence ordering for multi-document summarization. In: Proceedings of the 44th Annual Meeting of the Association of Computational Linguistics. (2006) 385–392
10. Carbonell, J., Goldstein, J.: The use of MMR, diversity-based reranking for reordering documents and producing summaries. In: Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. (1998) 335–336
11. Duboue, P.A., McKeown, K.R.: Empirically estimating order constraints for content planning in generation. In: Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics. (2001) 172–179
12. Elhadad, N., Mckeown, K.R.: Towards generating patient specific summaries of medical articles. In: Proceedings of the NAACL 2001 Workshop on Automatic Summarization. (2001)

13. Ji, P.D., Pulman, S.: Sentence ordering with manifold-based classification in multi-document summarization. In: Proceedings of Empherical Methods in Natural Language Processing. (2006) 526–533
14. Karamanis, N., Mellish, C.: Using a corpus of sentence orderings defined by many experts to evaluate metrics of coherence for text structuring. In: Proceedings of the 10th European Workshop on Natural Language Generation. (2005) 174–179
15. Madnani, N., Passonneau, R., Ayan, N.F., Conroy, J.M., Dorr, B.J., Klavans, J.L., O'Leary, D.P., Schlesinger, J.D.: Measuring variability in sentence ordering for news summarization. In: Proceedings of the 11th European Workshop on Natural Language Generation. (2007) 81–88
16. Mani, I., Schiffman, B., Zhang, J.: Inferring temporal ordering of events in news. In: Proceedings of North American Chapter of the ACL on Human Language Technology (HLT-NAACL 2003). (2003) 55–57
17. Mani, I., Wilson, G.: Robust temporal processing of news. In: The 38th Annual Meeting of the Association for Computational Linguistics. (2000) 69–76
18. McKeown, K.R., Klavans, J.L., Hatzivassiloglou, V., Barzilay, R., Eskin, E.: Towards multidocument summarization by reformulation: Progress and prospects. In: Proceedings of AAAI/IAAI. (1999) 453–460
19. Okazaki, N., Matsuo, Y., Ishizuka, M.: Improving chronological sentence ordering by precedence relation. In: Proceedings of the 20th International Conference on Computational Linguistics (COLING 04). (2004) 750–756
20. Radev, D.R., McKeown, K.R.: Generating natural language summaries from multiple on-line sources. Computational Linguistics **24** (1999) 469–500
21. Zhang, R., Li, W., Lu, Q.: Sentence ordering with event-enriched semantics and two- layered clustering for multi-document news summarization. In: Proceedings of COLING 2010. (2010) 1489–1497
22. Lapata, M.: Probablistic text structuring: Experiments with sentence ordering. In: Proceedings of the 41st Annual Meeting of the Association of Computational Linguistics. (2003) 542–552
23. Cristianini, N., Shawe-Taylor, J.: An Introduction to Support Vector Machines and Other Kernel-based Learning Methods. Cambridge University Press (2000)
24. Uchimoto, K., Murata, M., Ma, Q., Sekine, S., Isahara, H.: Word order acquisition from corpora. In: Proceedings of COLING 2000. (2000) 871–877
25. Kudoh, T.: TinySVM: Support Vector Machines. http://cl.aist-nara.ac.jp/ taku-ku/software/TinySVM/index.html (2000)

YUYA HAYASHI
TOTTORI UNIVERSITY,
4-101 KOYAMA-MINAMI, TOTTORI 680-8552, JAPAN
E-MAIL: <S082043@IKE.TOTTORI-U.AC.JP>

MASAKI MURATA
TOTTORI UNIVERSITY,
4-101 KOYAMA-MINAMI, TOTTORI 680-8552, JAPAN
E-MAIL: <MURATA@IKE.TOTTORI-U.AC.JP>

LIANGLIANG FAN
TOTTORI UNIVERSITY,
4-101 KOYAMA-MINAMI, TOTTORI 680-8552, JAPAN
E-MAIL: <K112001@IKE.TOTTORI-U.AC.JP>

MASATO TOKUHISA
TOTTORI UNIVERSITY,
4-101 KOYAMA-MINAMI, TOTTORI 680-8552, JAPAN
E-MAIL: <TOKUHISA@IKE.TOTTORI-U.AC.JP>

IJCLA VOL. 4, NO. 2, JUL-DEC 2013, PP. 169–188
RECEIVED 05/08/13 ACCEPTED 10/11/13 FINAL 30/11/13

Syntactic Dependency Based N-grams in Rule Based Automatic English as Second Language Grammar Correction

GRIGORI SIDOROV

Instituto Politécnico Nacional, Mexico

ABSTRACT

In this paper, we present a system for automatic English (L2) grammatical error correction. It participated in ConLL 2013 shared tasks. The system applies a set of simple rules for correction of grammatical errors. In some cases, it uses syntactic n-grams, i.e., n-grams that are constructed in a syntactic metric: namely, by following paths in dependency trees, i.e., there is special procedure that allows obtaining syntactic n-grams. Note that in general case syntactic n-grams permit introducing syntactic information into machine learning methods, because syntactic n-grams have all properties of traditional n-grams. The system is simple, practically does not use additional linguistic resources and was constructed in two months. Due to its simplicity it does not obtain better scores as compared to more sophisticated systems that use many resources, the Internet and machine learning methods, but it can be positioned as a baseline system for the task.

1 Introduction

The dominant paradigm in Computational Linguistics (CL) and Natural Language Processing (NLP) nowadays is based on machine learning methods. Most popular are supervised learning techniques because they obtain better results as compared with unsupervised approaches. The shortcoming of the supervised methods is the necessity of large linguistic data suitable for further application of supervised machine learning algorithms. In practice, it turns into the need of the large manually marked corpora. The problem is even bigger because each CL or NLP task needs a specific corpus marked in a unique manner. So, there should be as many different corpora as there are CL and NLP tasks, such as machine translation, automatic summarization, POS tagging, parsing, various levels of semantic and discourse annotation, etc.

On the other hand, an alternative to machine learning is the paradigm based on usage of human crafted rules. It is not so popular nowadays, though it was dominant until a couple of decades ago (until 90s) (Bolshakov, Gelbukh 2004). In this case, humans instead of annotating corpora are creating rules. It is obvious that for creating rules the humans try to take into account the same phenomena as machine learning algorithms. The current state of the art is that a machine learning algorithm can take into account so many textual (and especially contextual) features at many levels of language at the same time that it outperforms humans (Gelbukh 2013).

Interestingly, a growing interest to rule based approaches is related to a relatively new generative machine learning approaches such as Conditional Random Fields. These approaches use hand-crafted features that usually describe local context. It is known that Conditional Random Fields outperform traditional machine learning on classical tasks such as, for example, POS tagging. So, probably, a new paradigm is emerging that will be based both on machine learning algorithms and manually developed rules.

After this brief discussion about the tendency of use of rules in CL and NLP, let us describe the use of rules in the system presented in this paper. The problem discussed in the paper is related to the problem of automatic correction of grammatical errors of persons who are learning English as the second language (L2). Though various methods have been proposed for detecting and correcting such errors of different kinds: semantic errors (Bolshakov, Gelbukh 2003), malapropisms (Boshakov, Galicia-Haro, Gelbukh 2005), errors in lexical functions (Gelbukh, Kolesnivova 2013), the problem remains very relevant. In particular, this problem was represented in the ConLL 2013 shared task.

This paper describes the system that performs this task using a set of hand crafted rules. Some of these rules are based on the concept of syntactic dependency based n-grams (sn-grams), which we proposed recently (Sidorov, Velasquez, Stamatatos, Gelbukh & Chanona-Hernandez 2012, 2013, 2014; Sidorov 2013).

The proposed set of rules is simple and the whole development cycle of the system began about two months before the task deadline and took approximately only one person-month joint effort in total, which is relatively little effort. So it is not surprising that the system does not present excellent results, but instead due to its simplicity and quick development speed it can be positioned as a base line system for the task.

The rest of the paper is organized as follows. First, in Section 2 we describe the concept of syntactic dependency based n-grams that are used by our system. In Section 3 the ConLL shared task is described. After this in Section 4 we present our system and the rules, which it uses. The obtained scores are discussed in Section 5, and finally in Section 6 conclusions are drawn.

2 Syntactic Dependency Based N-grams

In this section we present briefly syntactic dependency based n-grams (syntactic n-grams, sn-grams). We introduced this concept in our previous works (Sidorov et al. 2012, 2013, 2014; Sidorov 2013). We have shown that application of syntactic n-grams gives better results than the use of traditional n-grams for the task of the authorship attribution. Similar idea was proposed in (Pado, Lapata 2007; Gelbukh 1999), but only as something very specific for certain tasks of syntactic or semantic analysis.

Note that sn-grams are not n-grams constructed using POS tags, as one may suppose just looking at the term. In fact, strictly speaking, it is wrong usage of the word "syntactic" because POS tags represent merely morphological data and syntactic information (context) is used only for disambiguation.

For explaining what for syntactic n-grams are used, we need to remind the reader the concept of the Vector Space Model (VSM). Majority of modern machine learning methods is based on Vector Space Model. The VSM is very versatile and can be used for characterization of any types of objects. General idea of the VSM is that any object in the world can be represented using certain features and these features have sets of possible values, so by choosing a set of features we define a VSM for the selected objects. Each object is represented by a vector of values of the selected features, i.e., it is a point in multidimensional vector space, being features the axes. Note that the features are ordered. Since we are talking about vectors, we can calculate their similarity in a formal way using, for example, the cosine measure. Once the VSM is constructed, all calculi are objective, but its construction is subjective: we can choose any features we like and scale the values in a manner we prefer.

Now, when speaking about texts, the features that are used typically for VSM construction are words or traditional n-grams—word sequences as they appear in texts. Usually, tf-idf values are used as values of these features. These values depend on the word or n-gram frequencies in texts.

There is modern research tendency that consists in reducing dimensions of the VSM using methods such as the Latent Semantic Analysis (LSA). It is possible because sets of vectors are equivalent to matrices, and the LSA is in practice an application of standard matrix processing technique−singular value decomposition (SVD).

The Vector Space Model representation is applied practically in any CL and NLP task with slight modifications.

Main criticism of the Vector Space Model is that it is purely statistical and does not reflect linguistic knowledge.

Our proposal is to introduce syntactic knowledge into the VSM by using other type of features, i.e., instead of traditional n-grams that are just sequences of words at the surface level, we suggest using syntactic n-grams that are obtained using linguistic (syntactic) knowledge, so that they reflect “real” relations between words.

The method of obtaining syntactic n-grams consists in following paths in syntactic tree and taking the words for n-grams in the order of their appearance. Obvious disadvantage of these features is that previous parsing is needed, but nowadays there are many freely available fast reliable parsers for many languages. We use dependency trees, but constituency trees can be used as well, because both types of trees reflect the same syntactic reality (Gelbukh, Calvo, Torres 2005).

In our previous works, we have proposed classification of syntactic n-gram types. Depending on the elements that constitute them, there can be syntactic n-grams of words / lemmas, POS tags, SR tags (names of Syntactic Relations), multiword expressions (Gelbukh, Kolesnikova 2013; Ledeneva, Gelbukh, García-Hernández 2008), and even of characters (Sidorov et al. 2012, 2013, 2014). There also can be mixed sn-grams, for example, one element is a POS tag and the other one is a lexical unit.

On the other hand, in (Sidorov, 2013) we have proposed to differentiate between continuous (non-interrupted path) and non-continuous (path with interruptions or returns) syntactic n-grams. The difference is that in case of continuous n-grams we follow the syntactic path as one continuous line, without interruption (returns, bifurcations), while in case of non-continuous n-grams the syntactic path can have interruptions (returns, bifurcations), and, thus, we can return to the same point in the tree. In the latter case special meta-language for syntactic n-gram representation is needed, because there can appear ambiguities. We proposed very simple meta-language with comas and brackets, which allows resolving the problem of ambiguities. It is clear that continuous syntactic n-grams is a special case of non-continuous sn-grams (with no interruptions/bifurcations/returns).

Now let us give some examples. We will use probably the most linguistically famous phrase by N. Chomsky "*Colorless green ideas sleep furiously*", where the words are used without any sense but the syntactic structure is maintained. Obviously, syntactic n-grams can be extracted from any phrase that we can parse.

Stanford parser produces the following output.

amod(ideas-3, colorless-1)
amod(ideas-3, green-2)
nsubj(sleep-4, ideas-3)
root(ROOT-0, sleep-4)
advmod(sleep-4, furiously-5)

Using this data we can construct the corresponding syntactic tree (Fig. 1) and then extract syntactic n-grams. First, let us consider only continuous (non-interrupted) n-grams. We start from the root and follow the arrows without returns. In case of word bigrams we have:

sleep ideas
ideas green
ideas colorless
sleep furiously

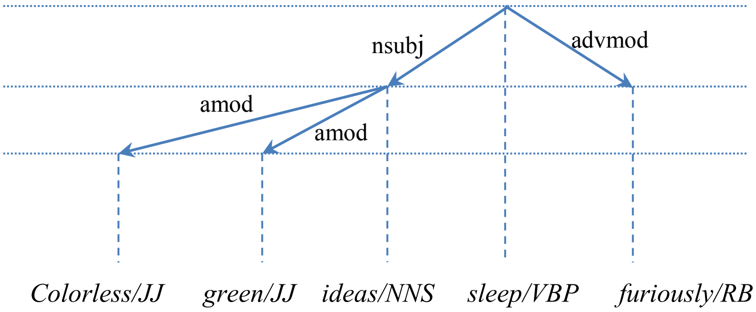


Fig. 1. Example of a dependency syntactic tree

Note that the head word is always the first element of sn-gram. If we compare it with traditional bigrams:

colorless green
green ideas
ideas sleep
sleep furiously

The obvious advantage is that instead of the traditional bigram of two adjectives "*colorless green*" we have the bigram "*ideas colorless*", which has much more sense.

Syntactic 3-grams of words are: *sleep ideas colorless*, *sleep ideas green*.

No more continuous syntactic n-grams of words can be constructed, but our tree is extremely simple. For more complex tree there are much more sn-grams.

We can also consider syntactic n-grams of POS tags, like bigrams *VBP-NNS*, *NNS-JJ*, *NNS-JJ*, *VBP-RB* or trigrams *VBP-NNS-JJ*, *VBP-NNS-JJ*.

Also syntactic n-grams of names of syntactic relations are possible, like *nsubj-amod*. Note that this type of n-grams does not exist for traditional n-grams.

Mixed syntactic n-grams are also possible, for example, if we mix POS tags and words, the following bigrams are extracted: *sleep-NNS*, *ideas-JJ*, *sleep-RB*, *VBP-ideas*, *VBP-furiously*, *NNS-green*, *NNS-colorless*. Also the following 3-grams: *sleep-NNS-JJ*, *sleep-ideas-JJ*.

VBP-NNS-green, *VPB-NNS-colorless*, *VBP-ideas-JJ*, *VBP-ideas-green*, *VBP-ideas-colorless*.

We can also mix SR tags (names of syntactic relations) with POS-tags or words/lemmas, for example, *nsubj-ideas-amod*, *VBP-nsubj-ideas-amod*, etc. In this case, it is a question of future experiments to determine which types of sn-grams give better results.

Now let us pass to non-continuous syntactic n-grams. In our tree, there is only two points of bifurcations: in *sleep* and *ideas*.

Note that in case of bigrams there is no distinction between continuous and non-continuous types.

The rules of the meta-language, which we have proposed for representation of non-continuous sn-grams, are simple: the elements of bifurcation are separated by comas (to distinguish them from a continuous path) and each bifurcation is taken in brackets. The rules are applied recursively. Extraction of these sn-grams can be performed by simple recursive algorithm (Sidorov, 2013).

Non-continuous syntactic 3-grams of words for the example sentence are:

sleep [*ideas*, *furiously*]
ideas [*colorless*, *green*]
sleep ideas colorless
sleep ideas green

There are two more non-continuous 3-grmas as compared to continuous 3-grams, which correspond exactly to bifurcations.

There are three 4-grams in the example.

sleep [*ideas* [*colorless, green*]]
sleep [*ideas colorless, furiously*]
sleep [*ideas green, furiously*]

Note that there is no coma in the first 4-gram between *ideas* and [*colorless, green*], because coma only separates elements of bifurcations.

There is also one non-continuous 5-gram.

sleep [*ideas* [*colorless, green*], *furiously*]

We hope that we have now explained the concept of syntactic n-gram and its types.

```
829 2 1 0    This          DT   1    nsubj   (ROOT(S(NP*)
829 2 1 1    caused        VBD -1    root    (VP*
829 2 1 2    problem       NN   1    dobj    (NP*)
829 2 1 3    like          IN   1    prep    (PP*
829 2 1 4    the           DT   5    det     (NP(NP*
829 2 1 5    appearance    NN   3    pobj    *)
829 2 1 6    of            IN   5    prep    (PP*
829 2 1 7    slums         NNS 6     pobj    (NP(NP*)
829 2 1 8    which         WDT16     dobj    (SBAR(WHNP*)
829 2 1 9    most          JJS  16   nsubj   (S(NP(NP*)
829 2 1 10   of            IN   9    prep    (PP*
829 2 1 11   the           DT   12   det     (NP*
829 2 112    time          NN   10   pobj    *)))
829 2 1 13   is            VBZ 16    cop     (VP*
829 2 1 14   not           RB   16   neg     *
829 2 1 15   safe          JJ   16   amod    (ADJP(ADJP*
829 2 1 16   due           JJ   7    rcmod   *)
829 2 1 17   to            TO   16   prep    (PP*
829 2 1 18   the           DT   20   det     (NP*
829 2 1 19   unhealthy     JJ   20   amod    *
829 2 1 20   environment   NN   17   pobj    *))))))))))
829 2 1 21   .             .    -    -       *))
```

Fig. 2. Example of a parsed sentence

3 ConLL Shared Task Description

ConLL Shared Task consists in the following. The training data was available for the registered teams. This data was processed previously by the Stanford parser (de Marneffe, MacCartney, Manning 2006). The data is part of the NUCLE corpus (Dahlmeier, Ng, Wu 2013). The data also contains the error types and the corrections of errors.

For example, the phrase "*This caused problem like the appearance of slums which most of the time is not safe due to the unhealthy environment*" is represented in the parsed variant as shown in Fig. 2.

The first four numbers correspond to the identifiers of the text, paragraph, sentence, and word correspondingly. Then the word itself comes together with its grammar tag (class). Three last columns contain syntactic data. The last column represents the constituency format that

```
<ANNOTATION>
<MISTAKE nid="829" pid="2" sid="1" start_token="2" end_token="3">
<TYPE>Nn</TYPE>
<CORRECTION>problems</CORRECTION>
</MISTAKE>
<MISTAKE nid="829" pid="2" sid="1" start_token="13" end_token="14">
<TYPE>Vform</TYPE>
<CORRECTION>are</CORRECTION>
</MISTAKE>
<MISTAKE nid="829" pid="2" sid="1" start_token="18" end_token="19">
<TYPE>ArtOrDet</TYPE>
<CORRECTION>their</CORRECTION>
</MISTAKE>
</ANNOTATION>
```

Fig. 3. Example of error information.

we, in our case, ignore. The remaining two columns contain the number of the head word (i.e., the word that is the head word for the current one) and the type of the syntactic relation.

The error information is presented in a separate file with XML encoding, see Fig. 3. Information about each error starts with the field <MISTAKE>, where the text, paragraph and sentence IDs are present, while *start_token* and *end_token* indicate position of the error in the sentence. The field <TYPE> contains the error type (see Section 3.1), and the field <CORRECTION> has the suggested correction of the error. For example, there are three errors in the example sentence as shown in Fig. 3. In our opinion, the corpus is a valuable resource because it contains manually annotated data, but it contains many polemic decisions, which can be seen in the example sentence. We would not consider as errors the words marked as the first and the third error. The suggested variants can be slightly preferred, but if they should be considered errors is not so clear.

The subjectivity in corpus preparation no doubt influences the final results of all systems during evaluation. The concept of what is an error should be defined clearer for more precise evaluation. We would suggest that some cases should be marked as “preferred correction” or “possible correction”. Later the systems that do not detect these cases should not be penalized, nor the systems that propose the possible corrections should not have any additional negative score, i.e., neither precision nor recall should be affected. In spite of these shortcomings,

Table 1. Statistics of grammatical errors in the data

Error type	Training data	%	Test data	%
Vform (Verb form)	1,451	9.1	122	7.4
SVA (Subject-verb agreement)	1,529	9.6	124	7.5
ArtOrDet (Article or determiner)	6,654	42.1	690	42.0
Nn (Noun number)	3,773	23.9	396	24.1
Prep (Preposition)	2,402	15.2	311	18.9
Total	15,809		1,643	

the effort of the organizers is valuable and should be highly appreciated.

After the period when the training data is available, the test data in the same format (but without error information) is released. The systems should correct errors in the test data. Special script in Python for evaluation is provided (Dahlmeier and Ng, 2012).

3.1 *Types of Errors Marked in the Data*

There are five types of errors considered in the task: noun number, subject-verb agreement, verb form, article/determiner and choice of prepositions.

Here we present examples of the error types.

First, let us see an example of the subject-verb agreement ("SVA" error type) error. In the phrase "*This endeavor to produce more nuclear power have stimulated the development of safer designs of nuclear reactors*." the auxiliary verb "*have*" should be changed to "*has*".

The other error type is related to use of prepositions ("Prep" error type). The following phrase "*These green house gases are the main cause to worldwide global warming which give rise to further catastrophes such as the rise in global temperature etc*." has an error in the preposition "*to*", which should be substituted by the preposition "*of*".

Error type caused by the wrong usage of a verb form ("Vform" error type) is present in the following sentence. "*Under this process, the attractiveness and practicality of the inventions will be improved such that they could be converted into useful products which accepted by most people*." Instead of the verb form "*accepted*", the form "*are accepted*" should be used.

The other error type is the incorrect choice of an article or a determiner (“ArtOrDet” error type), for example, in “*On one hand more and more virus and hack can access personal computers, so the secret data and documents may be stolen*.” the underlined article should be eliminated.

Finally, the last error is related to the wrong use of noun number (“Nn” error type). In the phrase “*Besides safety, convenience is also desirable for identifications.*” the word “*identification*” should be used in singular.

The errors statistics presented in Table 1 were calculated on the available data.

It can be observed that the test and training data are more or less proportional and the larger percentage of error types are “Article or Determiner” errors, followed by “Noun number” and “Preposition”. As usual, during the percentage calculus rounding effects can affect the total percentage value.

4 System Description

The system uses training data for construction of syntactic n-grams only (in this case they are used as syntactic patterns), and then apply a set of simple rules described below trying to detect each one of the five error types one after another in each sentence from the test data and correct them.

Error detection is done in certain order. We first process the possible “Noun number” errors, because if we process them later, then the errors in agreement are produced. If we want to correct these errors, we should also change the corresponding verb as far as its agreement is concerned. Fortunately, as the syntactic information is available, we can easily find the verb-noun (as the subject or part of the predicate) pairs.

4.1 *Linguistic Data Used by the System*

The system uses very few linguistic data, such as word lists, corpora or dictionaries.

First of all, it is necessary to mention that though the morphological data is present in the input sentence (parsed by the

Stanford parser), it is necessary to be able to perform morphological generation for producing the corrections of the errors. For this we need either English list of word forms with corresponding grammatical information and lemmas or algorithms of morphological analysis and generation. We used freely available word list from the FreeLing software (Padró, Collado, Reese, Lloberes, Castellón 2010). The list contains word forms, their grammar tag and the lemma for about 71,000 lemmas. Note that several grammar tags or even lemmas can correspond to the same word form, so the search should take them all into account.

...boarded board VBD board VBN
boarder boarder NN
boarders boarder NNS
boarding board VBG
boardroom boardroom NN
boardrooms boardroom NNS
boards board NNS board VBZ
boars boar NNS
boas boa NNS
boast boast NN boast VB boast VBP
boasted boast VBD boast VBN
boastful boastful JJ
boasting boast VBG
boasts boast NNS boast VBZ
boat boat NN boat VB boat VBP
boatbuilder boatbuilder NN
boatbuilders boatbuilder NNS
boated boat VBD boat VBN
boater boater NN
boaters boater NNS...

This list is ready for application to analysis, but if we need generation we should first reorder the list according to lemmas, and then, given a lemma and a grammar tag, find the corresponding word form.

For example, the reordered fragment of the list above contains:

...board NNS boards
board VBD boarded
board VBG boarding
board VBN boarded
board VBZ boards

boardroom NN boardroom
boardroom NNS boardrooms...

Note that if our morphological generator accepts a word form as the input, we should first apply morphological analysis for generation of the corresponding lemma, and only then call the generator.

Morphological generation is used during correction of the "Noun number", "Subject-Verb Agreement", and "Verb form" error types. It is not used in processing of the "Preposition" and "Article or Determiner" error types.

The other resource that we used is the list of uncountable nouns. The list of 250 most common uncountable nouns is available at www.englishclub.com > Learn English > Vocabulary > Nouns. For example,

...laughter
lava
leather
leisure
lightning
linguistics
literature
litter
livestock
logic
loneliness
love
luck
luggage
machinery
magic
mail
management
mankind
marble
mathematics
mayonnaise
measles
meat
metal
methane
milk
money
mud
music
nature
news
nitrogen
nonsense
nurture
nutrition
obedience
obesity
oil
oxygen
paper
passion
pasta
patience
permission
physics
poetry
pollution
poverty
power
pride
production
progress
pronunciation
psychology
publicity
punctuation
quality
quantity
quartz
racism
rain
recreation
relaxation
reliability
research
respect
revenge
rice
room
rubbish
rum
safety
salad
salt
sand
satire
scenery
seafood
seaside
shame
shopping
silence
sleep
smoke
smoking...

We used this list for checking the "Noun number" type of errors, when we consider that these nouns should not have the plural form.

Finally, we used the data provided for training by the organizers of the ConLL shared task, i.e., the sentences with syntactic data parsed by

Stanford parser. This data was used to extract syntactic n-grams that correspond to processing of the "Preposition" error type.

We used no other linguistic data. Some of the systems that participated in the task used vast corpora and Internet.

4.2 *Rules of the System*

As we mentioned before, first the "Noun number" error type is processed. We search the plural of the nouns from the list of uncountable nouns. If we find this situation, then we generate the noun in singular and change the verb (agreement) if this noun is a subject.

We made an exception for the noun "time" and do not consider it as uncountable, because its use in the common expressions such as "*many times*" is much more frequent than its use as an uncountable noun as in "*theory of time*" or "*what time is it now*?". Note that word sense disambiguation would be helpful in resolution of the mentioned ambiguities. In addition, the rule which considers the presence of the dependent words like "*many, a lot of, amount of*" could be added.

The next error type is the "Subject verb agreement". We use the very simple rule for verbs in present (with tags VB and VBZ): if its subject is a noun in singular or it is a third person singular pronoun (*he*, *she*, *it*) and the verb is not a modal verb then it should be the verb for third person singular (VBZ). If it is not so, then it is an error and we correct it changing VB to VBZ or vice versa and generating the corresponding verb form.

There are two additional rules for special situations. In case of coordinative construction in the subject we change the grammar number to plural. In case of one or several auxiliary verbs (marked as *aux* or *auxpass*), that auxiliary verb that has the smallest number in the sentence is considered, like, for example, in *have been doing*. This rule exploits fixed word order in English.

The "Verb form" error type includes vast and different types of errors, so we create rules only for some cases of this error type. The rules for verb form correction are as follows: 1) if we have a modal verb, then the depending verb should have a VB tag, 2) if we have an auxiliary verb "have", then the main verb should have a VB tag (perfect tense). We could have created more rules for treatment of situations like "to reforms → to reform", etc. These rules are necessary and would improve the performance, but they cover very small percentage of the data, so for the sake of time we omit further development in this direction.

The error type “Preposition” exploits the previously described concept of syntactic n-grams. In this case, we consider only continuous sn-grams, which are treated as syntactic patterns.

It is well-known that prepositions depend on lexical units that are their heads, for example, see (Eeg-Olofsson, Knutsson 2003), which has been used for, for example, syntactic disambiguation (Galicia-Haro, Gelbukh 2007; Calvo, Gelbukh 2003). In our case, we do not have enough training data, because we use just the available data of the task. So, the performance of our system will be limited to the repetitions of syntactic patterns in the test data.

We conducted several experiments and found out that it is worth considering the dependent word of the preposition as well. Due to very limited training data we are obliged to consider not the word itself, but its POS tag, otherwise our recall would be bad. Note that we consider the neighbors as obtained from the syntactic tree. This method of considering neighbors in syntactic path, instead of taking them directly from the text, corresponds to the previously discussed concept of syntactic n-grams. Here we are talking about mixed syntactic 3-grams. They are mixed because two elements are lexical units (words) and the third element is POS tag. These are continuous sn-grams because bifurcations are not considered.

The fragment of the data that we obtained from the training corpus is presented in Table 2. Total number of the obtained syntactic n-grams is 1,896.

Table 2. Format of the sn-gram data for processing of prepositions.

Wrong prep.	Right prep.	Head word (lemma)	Head word POS tag	Dep. word (lemma)	Dep. word POS tag
in	*for*	*risk*	*NN*	*disorder*	*NN*
from	*of*	*application*	*NN*	*RFID*	*NNP*
into	*	*develop*	*VBZ*	*disease*	*NN*
for	*to*	*advantage*	*NNS*	*human*	*NN*
for	*	*request*	*VBG*	*test*	*NN*

In Table 2, the first column contains the wrong preposition (the error), while the second column has the correct preposition, i.e., the correction. The asterisk corresponds to the absence of the preposition, i.e., the preposition should be deleted. The other columns contain normalized head word with its POS tag and normalized dependent word with its POS tag.

The continuous syntactic 3-grams, which correspond to the rows of the table, are: "*risk in NN → risk for NN*", "*application from NNP → application of NNP*", "*develop into NN → develop NN*", "*advantage for NN → advantage to NN*".

The rule which was implemented in the system is the following: if a relation with preposition is found, then take its head word, POS tag of the dependent word and search in the list of syntactic patterns. If the combination with all three elements is found, then change the preposition to the correct one.

In case of the errors related to "Article or Determiner" type, we only implemented the part related to (1) the choice of the allomorph "*a*" vs "*an*", and (2) the incompatibility of the article "*a*" with nouns in plural. All other rules related to these phenomena take into account discourse information, so they cannot be treated with simple context based rules, even using syntactic information.

5 Scores and Discussion

The results obtained with the evaluation script for our system (Ng, Wu, Wu, Hadiwinoto, Tetreault 2103) for the "SVA/Vform" error types are precision 8.13%, recall 12.42% and F1 measure 9.83%, which was the only error types considered. The results are not very high, though the results of the other systems are not much higher: the average scores are precision 11.82%, recall 20.89%, F1 measure 13.41%, while the best system got precision 17.89%, recall 38.94%, F1 measure 24.51%, but as we mentioned previously, our system uses a rule-based approach with very few additional resources, so it cannot compete with machine learning based approaches that additionally rely on vast lexical resources and the Internet.

Due to its simplicity, low use of additional resources, and very short development time, we position our system as a possible baseline system for the task.

On the other hand, we would like to mention that in some cases the rules should be used as a complementary technique for machine learning methods: don't guess if you know (Tapanainen, Voutilainen 1994). We consider that the following rules, which are exact, can complement machine learning systems: the rules for the article "a", the rules for uncountable nouns (in this case, word sense disambiguation would help to determine if the sense in the text is an uncountable noun or has some other use), the subject-verb agreement rules, the rules for correct verb form (here it should be mentioned that these rules cannot cover all errors, but only the most obvious cases).

It is always useful to perform an analysis of the errors committed by a system. Let us analyze the supposed errors committed by our system for the "Noun number" error type.

It performed 18 corrections, 3 of which coincide with the marks in the corpus data. Two of them are clear errors of the system: "*traffic jam*", where the word "*jam*" is used in a sense other than that of the "substance", and "*many respects*", where again the word "*respect*" has a different meaning to that of the uncountable noun. As we mentioned before, WSD techniques should be used to determine the correct sense.

There are 13 cases listed below (in the texts, the word "LIVINGS" is encountered 5 times the word and "QUANTITIES" is encountered two times), that our system marked as errors, because they are uncountable nouns in plural, but they are not marked in the corpus. Let us consider the nouns in capital letters:

peaceful(JJ) LIVINGS(NNS)...,
life(NN) QUALITIES(NNS)...,
Many(JJ) science(NN) FICTIONS(NNS)...,
does(VBZ) not(RB) have(VB) enough(JJ) LANDS(NNS)...,
indicates(VBZ) that(IN) the(DT) FOODS(NNS) the(DT) people(NNS) eat(VBP)...,
problem(NN) of(IN) public(JJ) TRANSPORTATIONS(NNS)...,
healthcare(NN) consume(VBP) large(JJ) QUANTITIES(NNS) of(IN) energy...,
this(DT) society(NN) may(MD) lack(VB) of(IN) LABOURS(NNS)...

Note that the words "equipment" and "usage" in plural were marked as errors in the corpus. In our opinion, it is inconsistent to mark these two as errors, and not to mark the other words from this list as such. While it is true that their use in plural is possible, it is clearly forced and is much less probable. At least, students of English should learn to use these words in singular only. Some of these mistakes (but not all) were corrected by the organizers for the final scoring data.

6 Conclusions

In this paper, we have described a system developed for the CoNLL-2013 shared task: automatic English (as second language, L2) grammar error correction.

The system relies on the rule-based approach. It uses very few additional linguistic data: a morphological analyzer and the list of 250 common uncountable nouns, along with the training data provided by the organizers.

The system uses the syntactic information available in the training data represented as syntactic n-grams, i.e., n-grams extracted by following the paths in dependency trees. These n-grams have certain advantages over traditional n-grams and allow introducing of syntactic information into machine learning.

The system is simple and was developed in a short period of time (2 months, 1 person/months). Since it does not employ any additional resources or sophisticated machine learning methods, it does not achieve high scores, but it could be considered as a baseline system for the task.

On the other hand, it shows what can be obtained using a simple rule-based approach and describes some situations when a rule-based approach can perform better than machine learning method.

ACKNOWLEDGEMENTS This work was done under partial support of the Mexican Government (CONACYT, SNI, COFAA-IPN, SIP-IPN 20120418, 20121823, 20131441, 20131702), CONACYT-DST India ("Answer Validation through Textual Entailment"), Mexico City Government (ICYT DF PICCO10-120), and FP7-PEOPLE-2010-IRSES: "Web Information Quality – Evaluation Initiative (WIQ-EI)" European Commission project 269180. I would like to thank Anubhav Gupta and Martin Tozer for help in implementing programming tasks; Dolors Catala, Angels Catena, and Sandrine Fuentes for help in formulating some rules of the system.

References

1. Bolshakov, I.A., Gelbukh, A.: Computational linguistics: Models, resources, applications. IPN–UNAM–FCE, (2004) 187 pp.
2. Bolshakov, I.A., Gelbukh, A.: Paronyms for Accelerated Correction of Semantic Errors. International Journal on Information Theories and Applications 10 (2003) 11–19
3. Bolshakov, I.A., Galicia-Haro, S.N., Gelbukh, A.: Detection and Correction of Malapropisms in Spanish by means of Internet Search. Lecture Notes in Artificial Intelligence 3658 (2005) 115–122
4. Calvo, H., Gelbukh, A. Improving Prepositional Phrase Attachment Disambiguation Using the Web as Corpus. Lecture Notes in Computer Science 2905 (2003) 604–610
5. Dahlmeier, D., Ng, H.T.: Better evaluation for grammatical error correction. In: Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL 2012) (2012) 568–572
6. Dahlmeier, D., Ng, H.T., Wu, S.M.: Building a large annotated corpus of learner English: The NUS corpus of learner English (2013)
7. de Marneffe, M.C., MacCartney, B. Manning, C.D.: Generating typed dependency parses from phrase structure parses. In: Proceedings of LREC 2006 (2006)
8. Eeg-Olofsson, J., Knutsson, O.: Automatic grammar checking for second language learners – the use of prepositions. In: Proceedings of NODALIDA’03 (2003)
9. Galicia-Haro, S.N., Gelbukh, A.: Web-Based Model for Disambiguation of Prepositional Phrase Usage. Lecture Notes in Artificial Intelligence 4827 (2007) 922–932
10. Gelbukh, A.: Natural language processing: Perspective of CIC-IPN. In: Proceedings of the International Conference on Advances in Computing, Communications and Informatics, ICACCI 2013, IEEE (2013) 2112–2121
11. Gelbukh, A.: Syntactic disambiguation with weighted extended subcategorization frames. In: Proceedings of PACLING-99, Pacific Association for Computational Linguistics, University of Waterloo, Canada (1999) 244–249
12. Gelbukh, A., Calvo, H. Torres, S.: Transforming a Constituency Treebank into a Dependency Treebank. Procesamiento de Lenguaje Natural 35 (2005) 145-152
13. Gelbukh, A., Kolesnikova, O.: Multiword Expressions in NLP: General Survey and a Special Case of Verb-Noun Constructions. In: Emerging Applications of Natural Language Processing: Concepts and New Research. IGI Global. (2013) 1–21

14. Gelbukh, A., Kolesnikova, O.: Semantic Analysis of Verbal Collocations with Lexical Functions. Studies in Computational Intelligence 414 (2013) 146 pp.
15. Ledeneva, Y., Gelbukh, A., García-Hernández, R.A.: Terms Derived from Frequent Sequences for Extractive Text Summarization. In: Proceedings of the International Conference on Intelligent Text Processing and Computational Linguistics, CICLing 2008. Lecture Notes in Computer Science 4919 (2008) 593–604
16. Ng, H.T., Wu, S.M., Wu, Y., Hadiwinoto, C., Tetreault, J.: The CoNLL-2013 Shared Task on Grammatical Error Correction. In: Proceedings of the Seventeenth Conference on Computational Natural Language Learning: Shared Task, Bulgary: ACL (2013).
17. Pado, S., Lapata, M.: Dependency-based construction of semantic space models. Computational Linguistics, 33(2) (2007) 161–199.
18. Padró, L., Collado, M., Reese, S., Lloberes, M., Castellón, I.: Freeling 2.1: Five years of open-source language processing tools. In: Proceedings of 7th Language Resources and Evaluation Conference (LREC 2010), ELRA (2010)
19. Sidorov, G., Velasquez, F., Stamatatos, E., Gelbukh, A., Chanona-Hernandez, L.: Syntactic dependency-based n-grams as classification features. Lecture Notes in Artificial intelligence 7630 (2012) 1–11.
20. Sidorov, G., Velasquez, F., Stamatatos, E., Gelbukh, A., Chanona-Hernandez, L.: Syntactic dependency-based n-grams: More evidence of usefulness in classification. In: Proceedings of the International Conference on Intelligent Text Processing and Computational Linguistics, CICLing 2013. Lecture Notes in Computer Science 7816 (2013) 13–24
21. Sidorov, G., Velasquez, F., Stamatatos, E., Gelbukh, A., Chanona-Hernandez, L. Syntactic N-grams as machine learning features for natural language processing. Expert Systems with Applications 41(3) (2014) 853–860 (to appear)
22. Sidorov, G.: Non-continuous Syntactic N-grams. Polibits 48 (2013) 67–75 (in Spanish, abstract and examples in English)
23. Tapanainen, P., Voutilainen, A.: Tagging accurately – don't guess if you know. In: Proceedings of ANLP '94 (1994)

GRIGORI SIDOROV
NATURAL LANGUAGE AND TEXT PROCESSING LABORATORY,
CENTRO DE INVESTIGACIÓN EN COMPUTACIÓN,
INSTITUTO POLITÉCNICO NACIONAL (IPN),
MEXICO CITY, MEXICO
WEB: <WWW.CIC.IPN.MX/~SIDOROV>

IJCLA VOL. 4, NO. 2, JUL-DEC 2013, PP. 189–202
RECEIVED 10/12/12 ACCEPTED 04/03/13 FINAL 13/03/13

Automatic Distinction between Natural and Automatically Generated Texts Using Morphological and Syntactic Information

LEONID CINMAN, PAVEL DYACHENKO, VADIM PETROCHENKOV, AND SVETLANA TIMOSHENKO

Institute for Information Transmission Problems, Russia

ABSTRACT

Our work lies in the field of automatic metrics for assessing text quality. However, the task we had to solve is different from the usual tasks of this domain. The traditional and most common formulation of the task is to distinguish well-written texts from poorly written ones, in which case it is presupposed that any text to be assessed is written by a human. Normally, the type of the text is also known: a scientific publication, news, etc. We set a more general task: to distinguish normal texts written by man, on one hand, from automatically generated texts or automatically processed and intentionally damaged natural texts, on the other hand. An additional difficulty is that "normal" texts in our collection contain lists, fragments of tables, and examples of bad texts with mistakes. We started by parsing our data with our syntactic parser for Russian, after which we trained an algorithm using words with extracted morphological and syntactic information. Our best results show 78.1% recall, 94.6% precision and 85.5% f-measure.

KEYWORDS: *Dependency parser, LibLinear, text quality, machine learning.*

1 Introduction

Our work lies in the field of automatic metrics for assessing text quality. Inside the domain we can see two streams of research – studies of readability and studies of coherence. The first one is presented, for example, in (Collins-Thompson, Callan, 2004), (Schwarm, Ostendorf, 2005). Papers by Barzilay, Lee (2004), and Soricut, Marcu (2006) can give an idea about the topics and methods in the second stream of studies. It is easy to see that while the researchers working on readability are focused on natural, human-written texts and their perception by other people, those who study text coherence work primarily with automatically generated texts. However, there are situations in which one has to process both automatically generated and human-written texts on the same principles: this will happen if the collection to be considered is heterogeneous.

To the best of our knowledge, there is only one recent paper dedicated to the uniform treatment of heterogeneous texts: (Louis, 2012). The author proposes to use genre-specific features to qualify texts, which means that at least we need to know beforehand what type of text we have – this is an indispensable condition for future treatment.

Our task, however, is different and simply formulated: we want to have an algorithm that could define whether a particular text is automatically generated (or automatically transformed from a natural text), or not. A simple question, but in a sense it may be considered as basic knowledge, which precedes any further processing.

An additional motivation for the experiment we are about to present is the situation in machine learning on Russian data. There is not much work done on Russian, besides, most of them report inferior performance for Russian than for English. There are many different explanations for this fact depending on the task. For example, Zagibalov, Belyatskaya, Carroll (2010) state the difference in precision and recall in the sentiment analysis task, and explain it by the fact that the way sentiment is expressed in Russian is different from how it is expressed in English. However, a closer look at the techniques used by the authors will show that Russian text was neither stemmed nor lemmatized. We believe that mediocre results for Russian in some NLP tasks can be explained by the lack of morphological analysis.

With our experiment, we hope to answer the following question: is general linguistic processing like lemmatizing and parsing of Russian data useful when they are prepared for machine learning, particularly in the task of rough assessment of text quality.

2 Corpus Description

The materials for our experiment were kindly provided by the Russian Internet company Yandex. As these materials are not freely distributed, we have to confine ourselves to a brief description and some examples.

We received a corpus of marked text fragments. Markup, performed semi-automatically, contains two tags, 0 and 1. 0 means that the text is good, while 1 means that the text is somehow damaged or unnatural. The subset of fragments marked with 1 shows a broad range of text distortions. The average length of the fragment is 2.5 sentences. The size of the corpus is 41594 fragments. Among them there are 5195 units labeled with 1, i.e. 12.5%.

Examples (1) to (2) are “bad” fragments, supplied with literal translations so that the reader can see the extent of badness:

(1) *Grif - ptica terpelivaja oshelomljon, uvidja eto, i sel i stal smotret na to, chto bylo voznikla kakaja-to okazalsja Dzhejms Hjedli Chejz. Grif - ptica terpelivaja tot stolik, chto prinadlezhal proroku Allaha Sulejmanu, synu Dauda.* ‘Griffon bird patient stunned seeing it, and sat down and began to look at what was appeared some was a James Hadley Chase. Griffon bird patient the table that belonged to the Prophet of Allah Suleiman, son of Daud.’

(2) *Posle etogo ol'ga neskol'ko s maloletnim hristom igorja narodnye svjatoslavom navisla vygodoj na drevljan, razgromiv ih.* ‘After that, Olga a few with young Christ igor folk with Svyatoslav hung on drevlyane as a profit, beating them’

Good fragments are exemplified by (3) and (4):

(3) *Poluchaetsja, chto my gotovy zaregistrirovat' Vam firmu za: 2600+2300+1100= 6 000 rub. III. Zatraty oposredovannye, t.e. kazhdyj opredeljaet dlja sebja sam, esli neobhodimo registrirovat' firmu: 1...7.Pechat' - 500 rub. 8. Kody statistiki - 700 rub.* ‘So we are ready to register your company for: 2600+2300+1100= 6000 Rubles. III. The costs are indirect, i.e. everybody decides for himself, in case that it is necessary to register a company, 1 ... 7. A stamp - 500 rubles. 8. Codes of statistics - 700 rubles’

(4) *Moe priobretenie Chery Tiggo, 4h4, 2,4. Polnyj komplekt, t.e. baza + kozha i ljuk. Poluchiv ee. poehala osvaivat' po prostoram Podmoskov'ja. Vse super!!* ‘My last purchase is Chery Tiggo, 4x4,

2.4. Full set, ie base + leather and sunroof. Receiving it, went to explore Moscow suburbs. It was great!'

Finally, the following example illustrates a special case of damaged text:

(5) *Gospodi, kak eto tak vdrug sovsem novyj mir nachalsja! No vse-taki, kak vy polagaete, vo vsem porechenkov ob jekstrasensah jetom nichego net osobenno ser'eznogo? Menja eto ochen' zanimaet. Skazhite, chem dokazhete vy mne, chto u vas budet luchshe?* 'God, this is so sudden that the entirely new world has begun! But still, do you think, Porechenkov about mediums there is nothing particularly serious there? I am very interested in this matter. Say, how will you prove to me that your world will be better?'

Obviously, in fragment (5), composed of three sentences, a Russian native speaker can easily identify the damaging section. Thus, "unnaturalness" may not span the whole fragment, and the right approach to this kind of damage is not to look for something in the general properties of the text, but to concentrate on the second sentence.

Considering the occurrence of such fragments, as well as the fact that our syntactic parser works mainly with individual sentences, not with the whole text, we manually refined the markup of the material. We have split all fragments into sentences. Each sentence coming from a "good" text was automatically marked with 0, whereas sentences received from the "bad" fragments were marked up as "bad" or "good" by a human annotator. In this way we compiled a corpus containing 115 331 sentences, of which 8543 were labeled with 1. In other words, we slightly changed the task from text quality assessment to sentence quality assessment.

3 ETAP-3 and The Parser for Russian

To obtain linguistic information, we used the multifunctional linguistic processor ETAP-3 (Boguslavsky et al., 2011). Its parsing module of Russian provides rich and diverse linguistic annotation. Many other Russian parsers yield a less detailed analysis. Some of them have evolved from the system ETAP-3 in a way: statistical parsers for Russian have been trained on the material of SynTagRus (Boguslavsky et al., 2009), a syntactically marked corpus of Russian Language, created with the help of ETAP-3.

The multifunctional ETAP-3 linguistic processor is a rule-based system able to execute several types of tasks, among them:

– a rule-based machine translation between Russian and English;
– synonymous and quasi-synonymous paraphrasing of sentences;
– automatic translation of natural language text into a semantic interlingua, UNL;
– identification of collocations in terms of lexical functions.

The parser performing syntactic analysis was elaborated as an auxiliary instrument for machine translation, but now it is often used independently.

To clarify what linguistic information we used for machine learning and where it comes from, a few words should be said about the parser's architecture.

The parser obtains the raw sentence as input and produces a dependency tree. Fig. 1 shows a dependency tree for sentence

(6) *Takim obrazom, v sovremennoj mirovoj ekonomike dejstvujut dve osnovnye tendentsii* 'Thus, two basic tendencies are present in modern world economy'

The nodes of the tree correspond to lemmas, which are supplied with morphological features, whilst the arcs are directed links labeled by names of syntactic relations. The parser makes use of about 65 different syntactic relations. Every link can be established by several rules which describe particular syntactic constructions. The algorithm first applies all possible rules to build all possible hypothetical links and then uses a variety of filters to delete excessive links so that the remaining ones form a dependency tree. Rules are divided into three groups: general rules, template rules and dictionary rules. The two latter types are evoked only if the sentence contains a word whose dictionary entry contains the respective rule or reference to the template rule. So, the ETAP syntax tunes itself to the lexical content of the sentence processed.

The ETAP-system utilizes a 120,000-strong Russian combinatorial dictionary, whose entries contain detailed descriptions of syntactic, semantic and combinatorial properties of words.

In the evaluation of the parser, SynTagRus is viewed as a gold standard. Evaluation results show the value of 0.900 for unlabeled

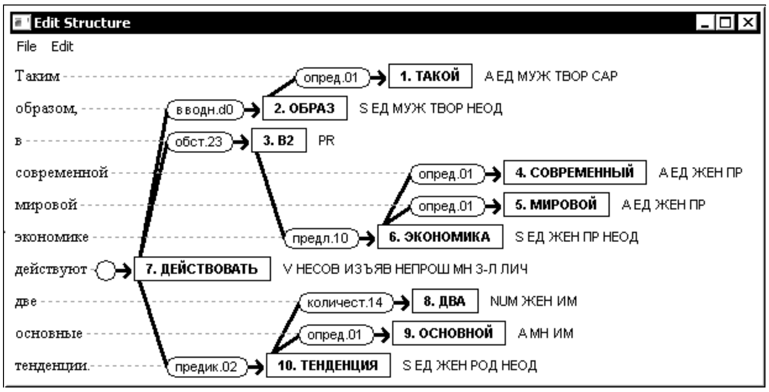


Fig. 1. The dependency tree for sentence (6)

attachment score, 0.860 for labeled attachment score, and 0.492 for unlabeled structure correctness.

For the cases when the parser fails to build an adequate syntactic tree, certain supplementary mechanisms are previewed. If the rules cannot produce a tree, some of the words in the sentence are linked by a soft-fail fictitious syntactic relation (see the pale link in Fig. 2, which gives a parse for an ungrammatical English sentence). When the parser finds a word that could not be found in the dictionary, this word is replaced by a suitable fictitious word (there are several types of such words, such as FICT-PERS or FICT-PLACE, which the parser attempts to substitute for unidentified proper names of people or locations) Normally, each node in the resulting tree corresponds to one word of the sentence parsed. Exceptions are cases where a word is a composite not assigned a dictionary entry (such as *vos'mitomnyj* 'eight-volume'), for which the parser produces two (or more) nodes in the dependency tree.

4 The Experiment

The first hypothesis we tested was that the damaged sentences have no standard structure so we can use fictitious syntactic links as direct markers of "bad" text. However, this hypothesis was not confirmed. "Good" and natural texts like (3) may turn out difficult for the parser

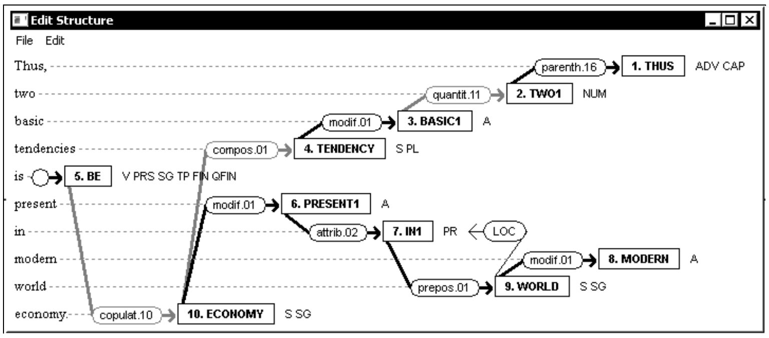


Fig. 2. The dependency tree for an ungrammatical sentence

due to many symbolic elements (numbers, +, = etc) which are likely cause errors. Within this approach we can only say that if the syntactic structures of the fragment do not contain any red link, it is highly probable that the fragment is “good”.

Assuming that a correlation between the linguistic features and the quality of text does exist, we designed an experiment with machine learning. From the syntactic tree, we extracted n-grams (n = 1, 2, 3) of:

- linearly adjacent wordforms,
- linearly adjacent lemmas,
- morphological feature sets arranged by linear order and by dependency order,
- syntactically connected wordforms,
- syntactically connected lemmas,
- syntactic relations that form a unidirectional path in the tree: we used consecutively arranged subtrees but no subtrees formed with sister nodes to get bigrams and trigrams of relations.

We also used as features generalized descriptions of subtrees which include morphological features and relations but no words (neither lemmas nor wordforms). For the complete list of features, see the Appendix below.

The feature set designed for machine learning was formed from all possible n-grams of different types listed above. For fragments we used n-grams extracted from all his sentences. Features in the set were not ordered. Feature set of every fragment was than transformed into a point in a multidimensional space and classified as 0 (“good” fragment) or 1 (“bad” fragment). We chose SVM, in particular linear SVM algorithm because of higher dimensions of our feature space (about 10^6). The practical implementation, that best fits our task is LibLinear library (Rong-En Fan et al., 2008), which shows good results on sparse data sets.

The first round of the experiment was to train the algorithm on marked fragments. 32,721 fragments formed the training set, and 8873 fragments were reserved for testing. In the testing set there were 1110 poorly written fragments, which amounts to 12.5%. The second round consisted in training the algorithm on sentences. The proportion of training /testing data remained the same. In absolute figures, we had 90,901 sentences in the training pool and the testing set contained totally 24,430 sentences, including 1814 “bad” units. It is easy to see that the part of “bad” stuff decreased to 7.42%. It is noteworthy that this decrease corresponds to the smaller proportion of “bad” sentences in the test sample, which is the effect of our re-tagging: after splitting the fragments we got some “good” sentences from bad fragments, but not vice versa.

First, we examined the relevance and effectiveness of types of n-grams mentioned above. Feature sets of every type (W, M, T, etc.) were tested separately, with widely varying regularization parameter C. In the next iteration we added to the characteristics that showed the greatest recall and f-measure (of all C) the set of n-grams of the second type (M + W, M + T, M + TL, etc.). When the recall no longer increase with the addition of regular types of characteristics, the feature selection was stopped. Our main goal was to maximize the recall, but it turned out that both recall and f-measure were maximized.

This experiment was done on the fragments, we did not repeat the procedure of the n-grams selection for the sentences. We used the set of features that proved to be the best in the fragments classification task.

5 Results

The procedure of the feature selection, described in Section 2, revealed that the best results can be obtained with the following set of characteristics: lemmas, syntactic relations, morphological feature sets corresponding to syntactically connected wordforms, wordforms (M + TL + TT + W in the Appendix and Table 1 below). These feature sets are listed in the descending order according to their contribution to the result. The training on the fragments shows the best result: 78.1%

Table 1. Best results for feature sets with and without syntactic information

	Fragments			Sentences		
	Recall	Preci-sion	Best f-measure	Recall	Preci-sion	Best f-measure
W+T+M	74.7%	95.5%	83.8%	64.4%	90.9%	75.4%
M+TL+TT+W	78.1%	94.6%	85.5%	65.3%	89.2%	75.4%

recall, 85.5% f-measure, 94.6% precision. The features based on lemmas give the most significant contribution to the result. While the system trained only on n-grams of wordforms shows 71.6% recall and 82.1% f-measure, the system trained on n-grams of lemmas perform 74.6% recall and 83.1 % f-measure.

It is also interesting to compare the best results obtained on fragments with the result obtained from a set of features, disregarding the features based on syntactic dependencies – lemmas, morphological feature sets arranged by linear order and wordforms (W + T + M in the table). The best result shown here is 74.7% recall, while f-measure is 83.8% and precision is 95.5%.

The above data show that the use of syntactic information allows a significantly improved recall in the text quality assessment task. The results of training on sentence data set were disappointing: they are much lower than the results for fragments (Table 1). However, they show the same pattern: additional information about the syntactic structure can improve the recall. We assume that the better performance of the fragment analyzer compared to the sentence analyzer can be explained as follows: the "bag" of features for the sentence is always smaller than the "bag" for the fragment.

These figures convince us that linguistic information, gathered without any supervision, even not 100% reliable, can make a remarkable contribution to the task of quality text assessment. Further experiments may refine the most relevant types of linguistic information or reveal other interesting correlations. We assume that it may be possible to benefit from sophisticated lexical information, such as semantic classes and syntactic frames.

6 Discussion

Notwithstanding the results, the experiment design and the approach in general have weak points of which we are fully aware.

It is well known that machine learning results strongly depend on the training data and their characteristics. Our experiment is no exception. The fragments of the collections were actually not intended for language processing, so there are artifacts in the good fragments that complicated their linguistic treatment and influenced the outcome of machine learning. E. g. some sentences are not reproduced in their original form, a few words in the middle are omitted and marked by the sign of ellipsis. This fact naturally holds true for our sentence markup. Having the imperfect data at the very beginning we could increase the uncertainty of some cases. We believe that the data gathered for this particular task could show better performance, but a new corpus is expensive to obtain.

To illustrate the weakest point of the approach, let us consider one more "bad" fragment:

Kak vyvesti zhirnoe pjatno? Pricheski dlja kruglogo lica. Gnevnyj harakter povyshaet status muzhchin, no diskreditiruet zhenchin. Razgnevannye zhenchiny proigryvajut v glazah publiki, togda kak razgnevannye muzhchiny, naoborot, zarabatyvajut dopolnitel'nye ochki. 'How to clean off a splodge? Hairstyles for round faces. The rage raises the status of men, but discredits women. Angry women lose in the public opinion while angry men earn extra points.'

This text is bad because the sentences are not coherent syntactic information has nothing to offer for the assessment of this kind of text: here we must resort to some text coherence metrics.

7 Conclusions

Our experiments have shown that general linguistic processing like lemmatization and parsing have a significant effect on the results of machine learning for the task of rough assessment of text quality . The experiments were held on Russian data, and we assume that for Russian and other inflexional languages such processing has a crucial importance. We also revealed the fact that syntactic information on sentence structure contributes to a higher recall. However, sentence quality assessment shows lower results than the text quality assessment. Further experiments could be focused on two different directions: we can study how parsing affects other types of machine learning tasks, e.g. sentiment detection, or investigate other types of linguistic information and their impact on the particular task of automatically generated/transformed text detection.

References

1. Barzilay, R., Lee, L.: Catching the drift: Probabilistic content models, with applications to generation and summarization. Proceedings of NAACL-HLT, 2004, pp. 113–120.
2. Boguslavsky, I., Iomdin, L., Timoshenko, S., Frolova, T.: Development of the Russian Tagged Corpus with Lexical and Functional Annotation. Metalanguage and Encoding Scheme Design for Digital Lexicography. MONDILEX Third Open Workshop. Proceedings, 2009, pp. 83-90.
3. Boguslavsky, I., Iomdin, L., Tsinman, L., Sizov, V., Petrochenkov, V.: Rule-Based Dependency Parser Refined by Empirical and Corpus Statistics. Proceedings of the International Conference on Dependency Linguistics, Depling'2011, 2011, pp. 318–327.
4. Collins-Thompson, K., Callan, J.: A language modeling approach to predicting reading difficulty. Proceedings of HLT-NAACL, 2004, pp. 193–200.
5. Louis, A.: Automatic Metrics for Genre-specific Text Quality. Proceedings of the NAACL-HLT Student Research Workshop, 2012, pp. 54–59.
6. Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, Chih-Jen Lin: LIBLINEAR: A Library for Large Linear Classification. Journal of Machine Learning Research 9, 2008, pp. 1871-1874.
7. Schwarm, Ostendorf, M.: Reading level assessment using support vector machines and statistical language models. Proceedings of ACL, 2005, pp. 523–530.
8. Soricut, R., Marcu, D.: Discourse generation using utility-trained coherence models. Proceedings of COLING-ACL, 2006, pp. 803–810.
9. Zagibalov, T., Belyatskaya K., Carroll, J.: Comparable English-Russian Book Review Corpora for Sentiment Analysis. Proceedings of the 1st Workshop on Computational Approaches to Subjectivity and Sentiment Analysis, 2010, pp. 67–72.

Appendix: Features Used

W1 a single wordform

W2 the pair of linearly adjacent wordforms (for the first and the last word we introduce an empty pair partner)

W3 the triple of linearly adjacent wordforms (for the first word we introduce two empty partners to form a triple, etc.)

M1 a single lemma

M2 the pair of linearly adjacent lemmas (for the first and the last word we introduce an empty pair partner)

M3 the triple of linearly adjacent lemmas (for the first word we introduce two empty partners to form a triple etc)

T1 a set of morphological features of a single word

T2 a pair of morphological feature sets corresponding to pair of linearly adjacent wordforms (with empty components for the first and the last wordform, respectively)

T3 a triple of morphological feature sets corresponding to triple of linearly adjacent wordforms (with empty components for the first and the last wordform, respectively)

TW2 a pair of wordforms connected with syntactic relation (with empty pair partners to the top and terminal nodes)

TW3 a triple of wordforms bound with syntactic relation in a serial way (with empty elements to the top and to the terminal node)

TM2 a pair of lemmas bound with syntactic relation (with empty pair partners to the top and terminal nodes)

TM3 a triple of lemmas bound with syntactic relation in a serial way (with empty elements to the top and to the terminal node)

TT2 a pair of morphological feature sets corresponding to the pair of syntactically bound wordforms (with empty pair partners for the first and the last wordforms, respectively)

TT3 a triple of morphological feature sets corresponding to triple of syntactically bound wordforms (with empty components for the first and the last wordforms, respectively)

TL1 a single syntactic relation

TL2 a pair of consecutive syntactic relations

TL3 a triple of consecutive syntactic relations

TTL2 a pair of morphological feature sets corresponding to the pair of syntactically connected wordforms and a syntactic relation itself (with empty elements for the top and the terminal nodes)

TTL3 a triple of morphological feature sets, corresponding to pair of syntactically connected wordforms and the binding syntactic relations (with empty elements for the top and the terminal nodes)

To give an example, for the subtree "in modern world economy" (Fig. 2) we have the following features:

W1 in, modern, world, economy

W2 (empty) – in, in – modern, modern – world, world – economy, economy – (empty)

W3 (empty) – (empty) – in, (empty) – in – modern, in – modern – world, modern – world – economy, world – economy – (empty), economy – (empty) – (empty)

M1[1] in, modern, world, economy

M2 (empty) – in, in – modern, modern – world, world – economy, economy – (empty)

M3 (empty) – (empty) – in, (empty) – in – modern, in – modern – world, modern – world – economy, world – economy – (empty), economy – (empty) – (empty)

T1 PR, A, S SG, S SG

T2 (empty) – PR, PR – A, A – S SG, S SG – S SG, S SG – (empty)

T3 (empty) – (empty) – PR, (empty) – PR – A, PR – A – S SG, A – S SG – S SG, S SG – S SG – (empty), S SG – (empty) – (empty)

TW2 (empty) – in, in – economy, economy – modern, economy – world, modern – (empty), world – (empty)

TW3 (empty) – (empty) – in, (empty) – in – economy, in – economy – modern, in – economy – world, economy – modern – (empty), economy – world – (empty), modern – (empty) – (empty), world – (empty) – (empty)

TM2 and TM3 repeat TW2 and TW3, respectively

TT2 (empty) – PR, PR – S SG, S SG – A, S SG – S SG, A – (empty), S SG – (empty)

TT3 (empty) – (empty) – PR, (empty) – PR – S SG, PR – S SG – A, PR – S SG – S SG, S SG – A – (empty), S SG – S SG – (empty), A – (empty) – (empty), S SG – (empty) – (empty)

TL1 prepos, modif, compos

[1] For English, the difference between the wordform and the lemma is minimal and can be seen only on the forms of plural for nouns and the tenses of verbs, but for inflexional languages such as Russian this difference is crucial, as discussed above.

TL2 (empty) – prepos, prepos – modif, prepos – compos, modif – (empty), compos – (empty)

TL3 (empty) – (empty) – prepos, (empty) – prepos – modif, (empty) – prepos – compos, prepos – modif – (empty), prepos – compos – (empty), modif – (empty) – (empty), compos – (empty) – (empty)

TTL2 (empty) – (empty) – PR, PR – prepos – S SG, S SG – modif – A, S SG – compos – S SG, A – (empty) – (empty), S SG – (empty) – (empty)

TTL3 (empty) – (empty) – PR, (empty) – PR – S SG, PR – S SG – A, PR – S SG – S SG, S SG – A – (empty), S SG – S SG – (empty), A – (empty) – (empty), S SG – (empty) – (empty)

Leonid Cinman
Institute for Information Transmission Problems,
Russian Academy of Sciences,
Bolshoy Karetny per. 19, Moscow, 127994, Russia
E-mail: <cinman@iitp.ru>

Pavel Dyachenko
Institute for Information Transmission Problems,
Russian Academy of Sciences,
Bolshoy Karetny per. 19, Moscow, 127994, Russia
E-mail: <pavelvd@iitp.ru>

Vadim Petrochenkov
Institute for Information Transmission Problems,
Russian Academy of Sciences,
Bolshoy Karetny per. 19, Moscow, 127994, Russia
E-mail: <vadim.petrochenkov@gmail.com>

Svetlana Timoshenko
Institute for Information Transmission Problems,
Russian Academy of Sciences,
Bolshoy Karetny per. 19, Moscow, 127994, Russia
E-mail: <timoshenko@iitp.ru>

Author Index

Editor-in-Chief

Editorial Board

Guest Editor of this Volume

Reviewing Committee of this Volume

Philipp Koehn
Valia Kordoni
Leila Kosseim
Mathieu Lafourcade
Krister Lindén
Elena Lloret
Bente Maegaard
Bernardo Magnini
Cerstin Mahlow
Sun Maosong
Katja Markert
Diana Mccarthy
Rada Mihalcea
Jean-Luc Minel
Ruslan Mitkov
Dunja Mladenic
Marie-Francine Moens
Masaki Murata
Preslav Nakov
Vivi Nastase
Costanza Navarretta
Roberto Navigli
Vincent Ng
Kjetil Nørvåg
Constantin Orasan
Ekaterina Ovchinnikova
Ted Pedersen
Viktor Pekar
Anselmo Peñas
Maria Pinango
Octavian Popescu
Irina Prodanof
James Pustejovsky
German Rigau
Fabio Rinaldi
Horacio Rodriguez
Paolo Rosso
Vasile Rus
Horacio Saggion
Franco Salvetti
Roser Sauri
Hinrich Schütze
Satoshi Sekine
Serge Sharoff
Grigori Sidorov
Kiril Simov
Vaclav Snasel
Thamar Solorio
Lucia Specia
Efstathios Stamatatos
Josef Steinberger
Ralf Steinberger
Vera Lúcia Strube De Lima
Mike Thelwall
George Tsatsaronis
Dan Tufis
Olga Uryupina
Karin Verspoor
Manuel Vilares Ferro
Aline Villavicencio
Piotr W. Fuglewicz
Annie Zaenen

Additional Referees for this Volume

Rodrigo Agerri
Katsiaryna Aharodnik
Ahmed Ali
Tanveer Ali
Alexandre Allauzen
Maya Ando
Javier Artiles
Noushin Rezapour Asheghi
Wilker Aziz
Vt Baisa

Alexandra Balahur
Somnath Banerjee
Liliana Barrio-Alvers
Adrián Blanco
Francis Bond
Dave Carter
Chen Chen
Jae-Woong Choe
Simon Clematide
Geert Coorman
Victor Darriba
Dipankar Das
Orphee De Clercq
Ariani Di Felippo
Maud Ehrmann
Daniel Eisinger
Ismail El Maarouf
Tilia Ellendorff
Milagros Fernández Gavilanes
Santiago Fernández Lanza
Daniel Fernández-González
Karën Fort
Sofia N. Galicia-Haro
Koldo Gojenola
Gintare Grigonyte
Francisco Javier Guzman
Masato Hagiwara
Kazi Saidul Hasan
Eva Hasler
Stefan Hoefler
Chris Hokamp
Stefan Höfler
Adrian Iftene
Iustina Ilisei
Leonid Iomdin
Pistol Ionut Cristian
Milos Jakubicek
Nattiya Kanhabua
Kurt Keena
Natalia Konstantinova
Vojtech Kovar
Kow Kuroda
Gorka Labaka
Shibamouli Lahiri
Egoitz Laparra
Els Lefever
Lucelene Lopes
John Lowe
Oier López de La Calle
Shamima Mithun
Tapabrata Mondal
Silvia Moraes
Mihai Alex Moruz
Koji Murakami
Vasek Nemcik
Zuzana Neverilova
Anthony Nguyen
Inna Novalija
Neil O'Hare
John Osborne
Santanu Pal
Feng Pan
Thiago Pardo
Veronica Perez Rosas
Michael Piotrowski
Soujanya Poria
Luz Rello
Francisco Ribadas-Pena
Tobias Roth
Jan Rupnik
Upendra Sapkota
Gerold Schneider
Djamé Seddah
Keiji Shinzato
João Silva
Sara Silveira
Sen Soori
Sanja Stajner
Tadej Štajner
Zofia Stankiewicz

Hristo Tanev
Irina Temnikova
Mitja Trampus
Diana Trandabat
Yasushi Tsubota
Srinivas Vadrevu
Josh Weissbock
Clarissa Xavier
Victoria Yaneva
Manuela Yapomo
Hikaru Yokono
Taras Zagibalov
Vanni Zavarella