In this paper, we have applied ROUGE-1 and ROUGE-2 which are simple n-gram measures. We compared our results with Microsoft, Mead Summarizer (Radev et al., 2003) and other two simple baselines: one which chooses 15% of words of the beginning of the judgment and second chooses last 10% of words of the judgment with human reference summaries. Both the baselines defined in this study are standard baselines for newspaper and research domains. The result shown in Table 3 highlights the better performances of our summarizer compared to other methods considered in this study. We can see that the results of MEAD and WORD summaries are not at the expected level, while our summarizer is best in terms of all four evaluation measures. Results are clearly indicated that our system performs significantly better than the other systems for legal judgments.

	MAP	F-measure	ROUGE-1	ROUGE-2
Baseline 1	0.370	0.426	0.522	0.286
Baseline 2	0.452	0.415	0.402	0.213
Microsoft Word	0.294	0.309	0.347	0.201
Mead	0.518	0.494	0.491	0.263
Our system	**0.646**	**0.654**	**0.685**	**0.418**

Table 3. MAP, F-measure and ROUGE scores.

4 Conclusion

This paper describes a novel method for generating a summary for legal judgments with the help of undirected graphical models. We observed that rhetorical role identification from legal documents is one of the primary tasks to understand the structure of the judgments. CRF model performs much better than rule based and other rule learning method in segmenting the text for legal domains. Our approach to summary extraction is based on the extended version of term weighting method. With the identified roles, the important sentences generated in the probabilistic model will be reordered or suppressed in the final summary. The evaluation results show that the summary generated by our summarizer is closer to the human generated head notes, compared to the other methods considered in this study. Hence the legal community will get a better insight without reading a full judgment. Moreover, our system-generated summary is more useful for lawyers to prepare the case history related to presently appearing cases.

Acknowledgement

We would like to thank the legal fraternity for the assistance and guidance governs to us. Especially we express our sincere gratitude to the advocates Mr. S.B.C. Karunakaran and Mr. K.N. Somasundaram for their domain advice and continuous guidance in understanding the structure of legal document and for hand annotated legal judgments.

References

Atefeh Farzindar and Guy Lapalme. 2004. *Legal text summarization by exploration of the thematic structures and argumentative roles,* In Text summarization Branches out workshop held in conjunction with ACL 2004, pages 27-34, Barcelona, Spain.

Atefeh Farzindar and Guy Lapalme. 2004. *Letsum, an automatic legal text summarizing system*, Legal Knowledge and Information System, Jurix 2004: The Seventeenth Annual Conference, Amsterdam, IOS Press, PP.11-18.

Ben Hachey and Claire Grover. 2005. *Sequence Modeling for sentence classification in a legal summarization system*, Proceedings of the 2005 ACM symposium on Applied Computing.

Bhatia, V.K., 1999. *Analyzing Genre: Language Use in Professional Settings*, London, Longman.

Cohen,W., and Singer, Y. 1999. *A simple, fast, and effective rule learner*, Proceedings of the Sixteenth National Conference on Artificial Intelligence (AAAI-99), AAAI Press, pp.335-342.

Dragomir Radev, Eduard Hovy, Kathleen McKeown. 2002. *Introduction to the special issue on summarization,* Computational Linguistics 28(4)4, Association for Computing Machinery.

Dragomir Radev, Jahna Otterbaher, Hong Qi, and Daniel Tam. 2003. *Mead Reducs: Michigan at DUC, 2003.* In DUC03, Edmonton, Alberta, Canada, May 31- June 1. Association for Computational Linguistics.

Dou Shen, Jian-Tao Sun, Hua Li, Qiang Yang, and Zheng Chen. 2007. *Document Summarization using Conditional Random Fields.* International Joint Conference on Artificial Intelligence, IJCAI 2007, Hyderabad, India, PP.2862-2867.

Friedmen, J.H., & and Popescu, B. E. 2005. *Predictive learning via rule ensembles* (Technical Report), Stanford University.

Fuchun Peng and Andrew McCullam, 2006. *Accurate information extraction from research papers using conditional random fields*, Information Processing Management, 42(4): 963-979.

John Lafferty, Andrew McCullam and Fernando Pereira, 2001. *Conditional Random Fields: Probabilistic models and for segmenting and labeling sequence data*, Proceedings of international conference on Machine learning.

Karen Sparck Jones and Julia Galliers. 1996. *Evaluating Natural Language Processing Systems: An Analysis and Review*. Natural Language Engineering, 4(2):175–190, Springer-Verlag.

Lin, Chin-Yew. 2004. *ROUGE: a Package for Automatic Evaluation of Summaries,* Proceedings of Workshop on Text Summarization, pp: 21--26, Barcelona, Spain.

Marie-Francine Moens, 2004. *An Evaluation Forum for Legal Information Retrieval Systems*? Proceedings of the ICAIL-2003 Workshop on Evaluation of Legal Reasoning and Problem-Solving Systems (pp. 18-24). International Organization for Artificial Intelligence and Law.

Saravanan , M., Ravindran, B. and Raman, S. 2006. *A Probabilistic Approach to Multi-document summarization for generating a Tiled Sumamry*, International Journal of Computational Intelligence and Applications, 6(2): 231-243, Imperial College Press.

Saravanan , M., Ravindran, B. and Raman, S. 2006. *Improving legal document Summarization using graphical models*, Legal Knowledge and Information System, JURIX 2006: The Nineteenth Annual Conference, Paris, IOS Press, PP.51-60.

Siegal, Sidney and N.John Jr. Castellan. 1988. *Nonparametric statistics for the behavioral sciences*, McGraw Hill, Berkeley, CA.

Simone Teufel and Marc Moens, 2002. *Summarizing scientific articles – experiments with relevance and rhetorical status,* Association of Computational Linguistics, 28(4): 409-445.

Yen-Yuan Yeh, Hao-Ren Ke, Wei-Pang Yang, and I-Heng Meng, 2005. *Text summarization using a trainable summarizer and latent semantic analysis*, Information processing management, 41(1):75-95.

Projection-based Acquisition of a Temporal Labeller

Kathrin Spreyer[*]
Department of Linguistics
University of Potsdam
Germany
spreyer@uni-potsdam.de

Anette Frank
Dept. of Computational Linguistics
University of Heidelberg
Germany
frank@cl.uni-heidelberg.de

Abstract

We present a cross-lingual projection framework for temporal annotations. Automatically obtained TimeML annotations in the English portion of a parallel corpus are transferred to the German translation along a word alignment. Direct projection augmented with shallow heuristic knowledge outperforms the uninformed baseline by 6.64% F_1-measure for events, and by 17.93% for time expressions. Subsequent training of statistical classifiers on the (imperfect) projected annotations significantly boosts precision by up to 31% to 83.95% and 89.52%, respectively.

1 Introduction

In recent years, supervised machine learning has become the standard approach to obtain robust and wide-coverage NLP tools. But manually annotated training data is a scarce and expensive resource. *Annotation projection* (Yarowsky and Ngai, 2001) aims at overcoming this resource bottleneck by scaling conceptually monolingual resources and tools to a multilingual level: annotations in existing monolingual corpora are transferred to a different language along the word alignment to a parallel corpus.

In this paper, we present a projection framework for *temporal annotations.* The TimeML specification language (Pustejovsky et al., 2003a) defines an annotation scheme for time expressions (*timex* for short) and events, and there are tools for the automatic TimeML annotation of English text (Verhagen et al., 2005). Similar rule-based systems exist for Spanish and Italian (Saquete et al., 2006). However, such resources are restricted to a handful of languages.

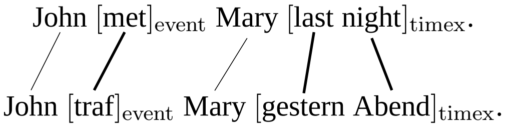

John [met]$_{\text{event}}$ Mary [last night]$_{\text{timex}}$.
John [traf]$_{\text{event}}$ Mary [gestern Abend]$_{\text{timex}}$.

Figure 1: Annotation projection.

We employ the existing TimeML labellers to annotate the English portion of a parallel corpus, and automatically project the annotations to the word-aligned German translation. Fig. 1 shows a simple example. The English sentence contains an event and a timex annotation. The event-denoting verb *met* is aligned with the German *traf*, hence the latter also receives the event tag. Likewise, the components of the multi-word timex *last night* align with German *gestern* and *abend*, respectively, and the timex tag is transferred to the expression *gestern abend*.

Projection-based approaches to multilingual annotation have proven adequate in various domains, including part-of-speech tagging (Yarowsky and Ngai, 2001), NP-bracketing (Yarowsky et al., 2001), dependency analysis (Hwa et al., 2005), and role semantic analysis (Padó and Lapata, 2006). To our knowledge, the present proposal is the first to apply projection algorithms to temporal annotations.

[*] The first author was affiliated with Saarland University (Saarbrücken, Germany) at the time of writing.

Cross-lingually projected information is typically noisy, due to errors in the source annotations as well as in the word alignment. Moreover, successful projection relies on the *direct correspondence assumption* (*DCA*, Hwa et al. (2002)) which demands that the annotations in the source text be homomorphous with those in its (literal) translation. The DCA has been found to hold, to a substantial degree, for the above mentioned domains. The results we report here show that it can also be confirmed for temporal annotations in English and German. Yet, we cannot preclude *divergence* from translational correspondence; on the contrary, it occurs routinely and to a certain extent systematically (Dorr, 1994). We employ two different techniques to filter noise. Firstly, the projection process is equipped with (partly language-specific) knowledge for a principled account of typical alignment errors and cross-language discrepancies in the realisation of events and timexes (section 3.2). Secondly, we apply aggressive data engineering techniques to the noisy projections and use them to train statistical classifiers which generalise beyond the noise (section 5).

The paper is structured as follows. Section 2 gives an overview of the TimeML specification language and compatible annotation tools. Section 3 presents our projection models for temporal annotations, which are evaluated in section 4. Section 5 describes how we induce temporal labellers for German from the projected annotations; section 6 concludes.

2 Temporal Annotation

2.1 The TimeML Specification Language

The TimeML specification language (Pustejovsky et al., 2003a)[1] and annotation framework emerged from the TERQAS workshop[2] in the context of the ARDA AQUAINT programme. The goal of the programme is the development of question answering (QA) systems which index content rather than plain keywords. Semantic indexing based on the identification of named entities in free text is an established method in QA and related applications. Recent years have also seen advances in relation extraction, a variant of event identification, albeit restricted in terms of coverage: the majority of systems addressing the task use a pre-defined set of—typically domain-specific—templates. In contrast, TimeML models events in a domain-independent manner and provides principled definitions for various event classes. Besides the identification of *events*, it addresses their relative ordering and anchoring in time by integrating *timexes* in the annotation. The major contribution of TimeML is the explicit representation of dependencies (so-called *links*) between timexes and events.

Unlike traditional accounts of events (e.g., Vendler (1967)), TimeML adopts a very broad notion of eventualities as "situations that happen or occur" and "states or circumstances in which something obtains or holds true" (Pustejovsky et al., 2003a); besides verbs, this definition includes event nominals such as *accident*, and stative modifiers (*prepared*, *on board*). Events are annotated with `EVENT` tags. TimeML postulates seven event classes: REPORTING, PERCEPTION, ASPECTUAL, I-ACTION, I-STATE, STATE, and OCCURRENCE. For definitions of the individual classes, the reader is referred to Saurí et al. (2005b).

Explicit timexes are marked by the `TIMEX3` tag. It is modelled on the basis of Setzer's (2001) `TIMEX` tag and the TIDES `TIMEX2` annotation (Ferro et al., 2005). Timexes are classified into four types: dates, times, durations, and sets.

Events and timexes are interrelated by three kinds of links: temporal, aspectual, and subordinating. Here, we consider only *subordinating links (slinks)*. Slinks explicate event modalities, which are of crucial importance when reasoning about the certainty and factuality of propositions conveyed by event-denoting expressions; they are thus directly relevant to QA and information extraction applications. Slinks relate events in modal, factive, counterfactive, evidential, negative evidential, or conditional relationships, and can be triggered by lexical or structural cues.

2.2 Automatic Labellers for English

The basis of any projection architecture are high-quality annotations of the source (English) portion

[1] A standardised version ISO-TimeML is in preparation, cf. Schiffrin and Bunt (2006).

[2] See `http://www.timeml.org/site/terqas/index.html`

$e \in E$	temporal entity
$l \in E \times E$	(subordination) link
$w_s \in W_s, w_t \in W_t$	source/target words
$al \in Al : W_s \times W_t$	word alignment
$A_s \ni a_s : E \to 2^{W_s}$	source annotation
$A_t \ni a_t :$	projected target
$(E \times A_s \times Al) \to 2^{W_t}$	annotation

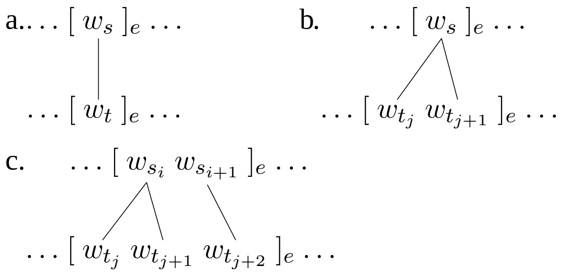
Table 1: Notational conventions.

of the parallel corpus. However, given that the projected annotations are to provide enough data for training a target language labeller (section 5), manual annotation is not an option. Instead, we use the TARSQI tools for automatic TimeML annotation of English text (Verhagen et al., 2005). They have been modelled and evaluated on the basis of the TimeBank (Pustejovsky et al., 2003b), yet for the most part rely on hand-crafted rules. To obtain a full temporal annotation, the modules are combined in a cascade. We are using the components for timex recognition and normalisation (Mani and Wilson, 2000), event extraction (Saurí et al., 2005a), and identification of modal contexts (Saurí et al., 2006).[3]

3 Informed Projection

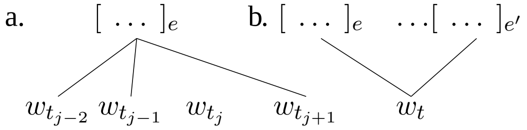
3.1 The Core Algorithm

Recall that TimeML represents temporal entities with `EVENT` and `TIMEX3` tags which are anchored to words in the text. Slinks, on the other hand, are not anchored in the text directly, but rather relate temporal entities. The projection of links is therefore entirely determined by the projection of the entities they are defined on (see Table 1 for the notation used throughout this paper): a link $l = (e, e')$ in the source annotation a_s projects to the target annotation a_t iff both e and e' project to non-empty sequences of words. The projection of the entities e, e' themselves, however, is a non-trivial task.



a.
$\ldots [\, w_s \,]_e \ldots$
$\ldots [\, w_t \,]_e \ldots$
b.
$\ldots [\, w_s \,]_e \ldots$
$\ldots [\, w_{t_j}\ w_{t_{j+1}} \,]_e \ldots$
c.
$\ldots [\, w_{s_i}\ w_{s_{i+1}} \,]_e \ldots$
$\ldots [\, w_{t_j}\ w_{t_{j+1}}\ w_{t_{j+2}} \,]_e \ldots$

Figure 2: Projection scenarios: (a) single-word 1-to-1, (b) single-word 1-to-many, (c) multi-word.



a.
$[\ \ldots\ ]_e$
$w_{t_{j-2}}$ $w_{t_{j-1}}$ w_{t_j} $w_{t_{j+1}}$
b.
$[\ \ldots\ ]_e$ $\ldots[\ \ldots\ ]_{e'}$
w_t

Figure 3: Problematic projection scenarios: (a) non-contiguous aligned span, (b) rivalling tags.

Given a temporal entity e covering a sequence $a_s(e)$ of tokens in the source annotation, the projection model needs to determine the extent $a_t(e, a_s, al)$ of e in the target annotation, based on the word alignment al. Possible projection scenarios are depicted in Fig. 2. In the simplest case (Fig. 2a), e spans a single word w_s which aligns with exactly one word w_t in the target sentence. In this case, the model predicts e to project to w_t. A single tagged word with 1-to-many alignments (as in Fig. 2b) requires a more thorough inspection of the aligned words. If they form a contiguous sequence, e can be projected onto the entire sequence as a multi-word unit. This is problematic in a scenario such as the one shown in Fig. 3a, where the aligned words do *not* form a contiguous sequence. There are various strategies, described in section 3.2, to deal with non-contiguous cases. For the moment, we can adopt a conservative approach which categorically blocks discontinuous projections. Finally, Fig. 2c illustrates the projection of an entity spanning multiple words. Here, the model composes the projection span of e from the alignment contribution of each individual word w_s covered by e. Again, the final extent of the projected entity is required to be contiguous.

With any of these scenarios, a problem arises when two distinct entities e and e' in the source an-

[3] TARSQI also comprises a component that introduces temporal links (Mani et al., 2003); we are not using it here because the output includes the entire tlink closure. Although Mani et al. (2006) use the links introduced by closure to boost the amount of training data for a tlink classifier, this technique is not suitable for our learning task since the closure might easily propagate errors in the automatic annotations.

```
project($a_s$, $al$):
    $a_{t,\mathsf{C}} = \emptyset$
    for each entity $e$ defined by $a_s$:
        $a_{t,\mathsf{C}}(e, a_s, al) = \bigcup^{\mathsf{C}}_{w_s \in a_s(e)} \mathsf{proj}(w_s, e, a_s, al)$
    for each link $l = (e, e')$ defined over $a_s$:
        if $a_{t,\mathsf{C}}(e, a_s, al) \neq \emptyset$ and $a_{t,\mathsf{C}}(e', a_s, al) \neq \emptyset$
        then define $l$ to hold for $a_{t,\mathsf{C}}$
    return $a_{t,\mathsf{C}}$
```

where

$$\begin{aligned}\mathsf{proj}(w_s, e, a_s, al) = \{w_t \in W_t \mid (w_s, w_t) \in al \wedge \\ \forall e' \in a_s.\ e' \neq e \Rightarrow w_t \notin a_{t,\mathsf{C}}(e', a_s, al)\}\end{aligned}$$

and

$$\bigcup^{\mathsf{C}} S = \begin{cases} \bigcup S & : \bigcup S \text{ is convex} \\ \emptyset & : \text{otherwise} \end{cases}$$

Figure 4: The projection algorithm.

notation have conflicting projection extents, that is, when $a_t(e, a_s, al) \cap a_t(e', a_s, al) \neq \emptyset$. This is illustrated in Fig. 3b. The easiest strategy to resolve conflicts like these is to pick an arbitrary entity and privilege it for projection to the target word(s) w_t in question. All other rivalling entities e' project onto their remaining target words $a_t(e', a_s, al) \setminus \{w_t\}$.

Pseudocode for this word-based projection of temporal annotations is provided in Fig. 4.

3.2 Incorporating Additional Knowledge

The projection model described so far is extremely susceptible to errors in the word alignment. Related efforts (Hwa et al., 2005; Padó and Lapata, 2006) have already suggested that additional linguistic information can have considerable impact on the quality of the projected annotations. We therefore augment the baseline model with several shallow heuristics encoding linguistic or else topological constraints for the choice of words to project to. Linguistically motivated filters refer to the part-of-speech (POS) tags of words in the target language sentence, whereas topological criteria investigate the alignment topology.

Linguistic constraints. Following Padó and Lapata (2006), we implement a filter which discards alignments to non-content words, for two reasons: (i) alignment algorithms are known to perform poorly on non-content words, and (ii) events as well as timexes are necessarily content-bearing and hence unlikely to be realised by non-content words. This *non-content (NC) filter* is defined in terms of POS tags and affects conjunctions, prepositions and punctuation. In the context of temporal annotations, we extend the scope of the filter such that it effectively applies to all word classes that we deem unlikely to occur as part of a temporal entity. Therefore, the NC filter is actually defined stronger for events than for timexes, in that it further blocks projection of events to pronouns, whereas pronouns may be part of a timex such as *jeden Freitag ‘every Friday’*. Moreover, events prohibit the projection to adverbs; this restriction is motivated by the fact that events in English are frequently translated in German as adverbials which lack an event reading (cf. head switching translations like *prefer to X* vs. German *lieber X ‘rather X’*). We also devise an unknown word filter: it applies to words for which no lemma could be identified in the preprocessing stage. Projection to unknown words is prohibited unless the alignment is supported bidirectionally. The strictness concerning unknown words is due to the empirical observation that alignments which involve such words are frequently incorrect.

In order to adhere to the TimeML specification, a simple transformation ensures that articles and contracted prepositions such as *am ‘on the’* are included in the extent of timexes. Another heuristics is designed to remedy alignment errors involving auxiliary and modal verbs, which are not to be annotated as events. If an event aligns to more than one word, then this filter singles out the main verb or noun and discards auxiliaries.

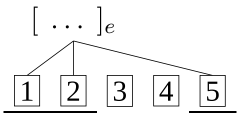
Topological constraints. In section 3.1, we described a conservative projection principle which rejects the transfer of annotations to non-contiguous sequences. That model sets an unnecessarily modest upper bound on recall; but giving up the contiguity requirement entirely is not sensible either, since it is indeed highly unlikely for temporal entities to be realised discontinuously in either source or target language (*noun phrase cohesion*, Yarowsky and Ngai (2001)). Based on these observations, we propose two refined models which manipulate the projected annotation span so as to ensure contiguity. One model identifies and discards *outlier alignments*, which actively violate contiguity; the other one adds *missing alignments*, which form gaps. Technically, both models establish convexity in non-convex sets. Hence, we first have to come up with a backbone model which is less restrictive than the baseline, so that the convexation models will have a basis to operate on. A possible backbone model $a_{t,0}$ is provided in (1).

(1) $$a_{t,0}(e, a_s, al) = \bigcup_{w_s \in a_s(e)} \mathrm{proj}(w_s, e, a_s, al)$$

This model simply gathers all words aligned with any word covered by e in the source annotation, irrespective of contiguity in the resulting sequence of words. Discarding outlier alignments is then formalised as a reduction of $a_{t,0}$'s output to (one of) its greatest convex subset(s) (GCS). Let us call this model $a_{t,\mathrm{GCS}}$. In terms of a linear sequence of words, $a_{t,\mathrm{GCS}}$ chooses the longest contiguous subsequence. The GCS-model thus serves a filtering purpose similar to the NC filter. However, whereas the latter discards single alignment links on linguistic grounds, the former is motivated by topological properties of the alignment as a whole.

The second model, which fills gaps in the word alignment, constructs the *convex hull* of $a_{t,0}$ (cf. Padó and Lapata (2005)). We will refer to this model as $a_{t,\mathrm{CH}}$. The example in (2) illustrates both models.

(2)



$\bigcup^C$: $\emptyset$
GCS : $\{1, 2\}$
CH : $\{1, 2, 3, 4, 5\}$

Here, entity e aligns to the non-contiguous token sequence $[1, 2, 5]$, or equivalently, the non-convex set $\{1, 2, 5\}(= a_{t,0}(e))$. The conservative baseline $a_{t,\mathrm{C}}$ rejects the projection altogether, whereas $a_{t,\mathrm{GCS}}$ projects to the tokens 1 and 2. The additional padding introduced by the convex hull ($a_{t,\mathrm{CH}}$) further extends the projected extent to $\{1, 2, 3, 4, 5\}$.

Alignment selection. Although bi-alignments are known to exhibit high precision (Koehn et al., 2003), in the face of sparse annotations we use unidirectional alignments as a fallback, as has been proposed in the context of phrase-based machine translation (Koehn et al., 2003; Tillmann, 2003). Furthermore, we follow Hwa et al. (2005) in imposing a limit on the maximum number of words that a single word may align to.

4 Experiments

Our evaluation setup consists of experiments conducted on the English-German portion of the Europarl corpus (Koehn, 2005); specifically, we work with the preprocessed and word-aligned version used in Padó and Lapata (2006): the source-target and target-source word alignments were automatically established by GIZA++ (Och and Ney, 2003), and their intersection achieves a precision of 98.6% and a recall of 52.9% (Padó, 2007). The preprocessing consisted of automatic POS tagging and lemmatisation.

To assess the quality of the TimeML projections, we put aside and manually annotated a development set of 101 and a test set of 236 bisentences.[4] All remaining data (approx. 960K bisentences) was used for training (section 5). We report the weighted macro average over all possible subclasses of timexes/events, and consider only exact matches. The TARSQI annotations exhibit an F_1-measure of 80.56% (timex), 84.64% (events), and 43.32% (slinks) when evaluated against the English gold standard.

In order to assess the usefulness of the linguistic and topological parameters presented in section 3.2, we determined the best performing combination of parameters on the development set. Not surprisingly, event and timex models benefit from the various heuristics to different degrees. While the projection of events can benefit from the NC filter, the projection of timexes is rather hampered by it. Instead, it exploits the flexibility of the GCS convexation model together with a conservative limit of 2 on per-word alignments. In the underlying data sample of 101 sentences, the English-to-German alignment direction appears to be most accurate for timexes. Table 2 shows the results of evaluating the optimised models on the test set, along with the baseline from section 3.1 and a "full" model which activates all

[4] The unconventional balance of test and development data is due to the fact that a large portion of the annotated data became available only after the parameter estimation phase.

	events			slinks			time expressions		
model	prec	recall	F	prec	recall	F	prec	recall	F
timex-optimised	48.53	33.73	39.80	30.09	10.71	15.80	**71.01**	**52.76**	**60.54**
event-optimised	**50.94**	**44.23**	**47.34**	30.96	14.29	19.55	56.55	42.52	48.54
combined	**50.98**	**44.36**	**47.44**	**30.96**	**14.29**	**19.55**	**71.75**	**52.76**	**60.80**
baseline	52.26	33.46	40.80	26.98	10.71	15.34	49.53	37.80	42.87
full	51.10	40.42	45.14	29.95	13.57	18.68	73.74	54.33	62.56

Table 2: Performance of projection models over test data.

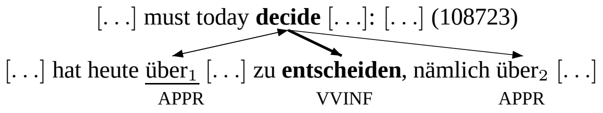

[. . .] must today **decide** [. . .]: [. . .] (108723)
[. . .] hat heute über$_1$ [. . .] zu **entscheiden**, nämlich über$_2$ [. . .]
APPR
VVINF
APPR

Figure 5: Amending alignment errors.

	event		timex	
data	prec	recall	prec	recall
all	53.15	45.14	73.74	53.54
best 75%	54.81	47.06	74.61	62.82

Table 3: Correlation between alignment probability and projection quality.

heuristics. The results confirm our initial assumption that linguistic and topological knowledge does indeed improve the quality of the projected annotations. The model which combines the optimal settings for timexes and events outperforms the uninformed baseline by 17.93% (timexes) and 6.64% (events) F_1-measure. However, exploration of the model space on the basis of the (larger and thus presumably more representative) test set shows that the optimised models do not generalise well. The *test set*-optimised model activates all linguistic heuristics, and employs $a_{t,\mathrm{CH}}$ convexation. For events, projection considers bi-alignments with a fallback to unidirectional alignments, preferably from English to German; timex projection considers all alignment links. This test set-optimised model, which we will use to project the training instances for the maximum entropy classifier, achieves an F_1-measure of 48.82% (53.15% precision) for events and 62.04% (73.74% precision) for timexes.[5]

With these settings, our projection model is capable of repairing alignment errors, as shown in Fig. 5, where the automatic word alignments are represented as arrows. The conservative baseline considering only bidirectional alignments discards all alignments but the (incorrect) one to *über*$_1$. The optimised model, on the other hand, does not exclude any alignments in the first place; the faulty alignments to *über*$_1$ and *über*$_2$ are discarded on linguistic grounds by the NC filter, and only the correct alignment to *entscheiden* remains for projection.

5 Robust Induction

The projected annotations, although noisy, can be exploited to train a temporal labeller for German. As Yarowsky and Ngai (2001) demonstrate for POS tagging, aggressive filtering techniques applied to vast amounts of (potentially noisy) training data are capable of distilling relatively high-quality data sets, which may then serve as input to machine learning algorithms. Yarowsky and Ngai (2001) use the Model-3 alignment score as an indicator for the quality of (i) the alignment, and therefore (ii) the projection. In the present study, discarding 25% of the sentences based on this criterion leads to gains in both recall and precision (Table 3). In accordance with the TimeML definition, we further restrict training instances on the basis of POS tags by basically re-applying the NC filter (section 3.2). But even so, the proportion of positive and negative instances remains heavily skewed—an issue which we will address below by formulating a 2-phase classi-

[5] The model actually includes an additional strategy to adjust event and timex class labels on the basis of designated FrameNet frames; the reader is referred to Spreyer (2007), ch. 4.5 for details.