Analyzing Short-Answer Questions and their Automatic Scoring

Studies on Semantic Relations in Reading Comprehension and the Reduction of Human Annotation Effort

Andrea Horbach

Dissertation zur Erlangung des Grades
des Doktors der Philosophie
an der Philosophischen Fakultät
der Universität des Saarlandes

Saarbrücken
2019

Dekan	Prof. Dr. Heinrich Schlange-Schöningen
Berichterstatter/innen	Prof. Dr. Manfred Pinkal
	Prof. Dr. Elke Teich
	Prof. Dr. Alexis Palmer
Tag der letzten Prüfungsleistung	23.10.2018

Acknowledgments

I want to thank the people who made this thesis possible.

My supervisors Manfred Pinkal and Alexis Palmer constantly offered advice, support and encouragement throughout the whole process of this thesis. Elke Teich accompanied the final steps as one of my reviewers.

Thanks go to all my colleagues in Saarbrücken for many fruitful discussions and moral support. Three people stood out in particular: Stefan Thater always had an open ear for me and was usually the one to sanity check any new idea. Annemarie Friedrich also discussed ideas with me and especially was a tremendous help in shaping conference papers. Diana Steffen was there for me from my very first day on, offering support in smaller and bigger crises.

With Magdalena Wolska, I worked on the Allegro project, where we collected the Laempel data set together.

Frau Kröner, as well as Christoph Clodo and the whole system administration group helped me with any administrative or technical problems.

I would also like to thank the students I was fortunate to work with and whose theses I co-supervised: Simon Ostermann, Nikolina Koleva, Jakob Prange (all three of which became my colleagues later on), Jana Ott, Jonathan Poitz, Lena Keiper and Stefan Ecker. The work with Simon, Nikolina and Lena also contributed to this thesis.

During my years in Saarbrücken, we had several meetings with Detmar Meurers and his group in Tübingen, from which I got valuable feedback and advice.

I am also grateful to Torsten Zesch, who got me involved in the INDUS network. I was fortunate to get in touch with many inspiring people from the area of language learning and benefit from our biannual network meetings. Torsten also gave me the opportunity to finish my thesis while I was already working with him on new projects.

I want to thank my proofreaders: Daniela Daniel, Oliver Vogel, and Victoria Issacs.

And finally, this thesis would not have been possible without the endless support from my family: Mama, Papa, thank you for always being there for me. Matze, I wouldn't know what to do without you.

Abstract

Short-answer questions are a wide-spread exercise type in many educational areas. Answers given by learners to such questions are scored by teachers based on their content alone ignoring their linguistic correctness as far as possible. They typically have a length of up to a few sentences. Manual scoring is a time-consuming task, so that automatic scoring of short-answer questions using natural language processing techniques has become an important task.

This thesis focuses on two aspects of short-answer questions and their scoring: First, we concentrate on a reading comprehension scenario for learners of German as a foreign language, where students answer questions about a reading text. Within this scenario, we examine the multiple relations between reading texts, learner answers and teacher-specified target answers. Second, we investigate how to reduce human scoring workload by both fully automatic and computer-assisted scoring. The latter is a scenario where scoring is not done entirely automatically, but where a teacher receives scoring support, for example, by means of clustering similar answers together.

Addressing the first aspect, we conduct a series of corpus annotation studies which highlight the relations between pairs of learner answers and target answers, as well as between both types of answers and the reading text they refer to. We annotate sentences from the reading text that were potentially used by learners or teachers for constructing answers and observe that, unsurprisingly, most correct answers can easily be linked to the text; incorrect answers often link to the text as well, but are often backed up by a part of the text not relevant to answer the question. Based on these findings, we create a new baseline scoring model which considers for correctness whether learners looked for an answer in the right place or not.

After identifying those links into the text, we label the relation between learner answers and target answers as well as between reading texts and answers by annotating entailment relations. In contrast to the widespread assumption that scoring can be fully mapped to the task of recognizing textual entailment, we find the two tasks to be only closely related and not completely equivalent. Correct answers do often, but not always, entail the target answer, as well as part of the related text, and incorrect answers do most of the time not stand in an entailment relation to the target answer, but often have some overlap with the text.

This close relatedness allows us to use gold-standard entailment information to improve the performance of automatic scoring. We also use links between learner answers and both reading texts and target answers in a statistical alignment-based scoring approach using methods from machine translation and reach a performance comparable to an existing knowledge-based alignment approach.

Our investigations into how human scoring effort can be reduced when learner answers are

manually scored by teachers are based on two methods: active learning and clustering. In the active learning approach, we score particularly informative items first, i.e., items from which a classifier can learn most, identifying them using uncertainty-based sample selection. In this way, we reach a higher performance with a given number of annotation steps compared to randomly selected answers. In the second research strand, we use clustering methods to group similar answers together, such that groups of answers can be scored in one scoring step. In doing so, the number of necessary labeling steps can be substantially reduced.

When comparing clustering-based scoring to classical supervised machine learning setups, where the human annotations are used to train a classifier, supervised machine learning is still in the lead in terms of performance, whereas clusters provide the advantage of structured output. However, we are able to close part of the performance gap by means of supervised feature selection and semi-supervised clustering.

In an additional study, we investigate the automatic processing of learner language with respect to the performance of part-of-speech (POS) tagging tools. We manually annotate a German reading comprehension corpus both with spelling normalization and POS information and find that the performance of automatic POS tagging can be improved by spell-checking the data using the reading text as additional evidence for lexical material intended in a learner answer.

Zusammenfassung

Short-Answer-Fragen sind ein weit verbreiteter Aufgabentyp in vielen Bildungsbereichen. Die Antworten, die Lerner zu solchen Aufgaben geben, werden von Lehrenden allein auf Grundlage ihres Inhalts bewertet; linguistische Korrektheit wird soweit möglich ignoriert.

Diese Doktorarbeit legt ihren Schwerpunkt auf zwei Aspekte im Zusammenhang mit Short-Answer-Fragen und ihrer Bewertung: Zum einen betrachten wir ein Leseverständnisszenario, bei dem Studenten Fragen zu Lesetexten beantworten. Dabei untersuchen wir insbesondere die verschiedenen Beziehungen, die es zwischen Lesetexten, Lernerantworten und vom Lehrer erstellten Musterantworten gibt. Zum anderen untersuchen wir, wie der menschliche Bewertungsaufwand durch voll-automatisches und computergestütztes Bewerten reduziert werden kann. Bei letzterem handelt es sich um ein Szenario, in dem Lehrer bei der Bewertung unterstützt werden, z.B. indem ähnliche Antworten automatisch gruppiert werden.

Zur Untersuchung des ersten Aspekts unternehmen wir eine Reihe von Korpusannotationsstudien, die sowohl die Beziehungen zwischen Lerner- und Musterantworten beleuchten, als auch die Beziehung zwischen diesen Antworten und dem Lesetext, auf den sie sich beziehen. Wir annotieren Sätze aus dem Lesetext, die vermutlich bei der Formulierung einer Antwort benutzt wurden und machen die zu erwartende Beobachtung, dass die meisten korrekten Antworten problemlos mit bestimmten Textpassagen in Verbindung gebracht werden können. Inkorrekte Antworten haben ebenfalls oft eine Verbindung zu bestimmten Textpassagen, die aber oft für die jeweilige Frage nicht relevant sind. Auf Grundlage dieser Erkenntnisse entwerfen wir ein neues Baseline-Bewertungsmodell, das für die Korrektheit einer Antwort nur in Betracht zieht, ob der Lerner die Antwort an der richtigen Stelle im Lesetext gesucht hat oder nicht.

Nachdem wir diese Verbindungen in den Text identifiziert haben, annotieren wir die Relation zwischen Lerner- und Musterantworten und zwischen Texten und Antworten mit Entailment-Relationen. Im Gegensatz zur der weitverbreiteten Annahme, dass das Bewerten von Short-Answer-Fragen und das Erkennen von Textual-Entailment-Relationen zwischen Lerner und Musterantworten sich direkt entsprechen, finden wir heraus, dass die beiden Aufgaben nur nahe verwandt aber nicht vollständig äquivalent sind. Korrekte Antworten entailen meistens, aber nicht immer, die Musterantwort und auch den entsprechenden Satz im Lesetext. Inkorrekte Antworten stehen meist in keiner Entailmentrelation mit der Musterantwort, haben aber oft zumindest teilweisen Overlap mit dem Text.

Diese nahe Verwandtschaft erlaubt es uns, Goldstandard-Entailmentinformation zu benutzen, um die Performanz beim automatischen Bewerten zu verbessern. Wir benutzen die annotierten Verbindungen zwischen Lesetexten und Antworten auch in einem Scoringansatz, der auf statistischem Alignment basiert und Methoden aus dem Bereich der maschinellen Übersetzung nutzt.

Dabei erreichen wir eine Scoringgenauigkeit, die mit Ansätzen, die ein existierendes wissensbasiertes Alignment nutzen, vergleichbar ist.

Unsere Untersuchungen, wie der Bewertungsaufwand beim Menschen verringert werden kann, wenn Antworten vom Lehrer manuell bewertet werden, basieren auf zwei Methoden: Active Learning und Clustering. Beim Active-Learning-Ansatz werden besonders informative Antworten vorrangig zur Bewertung ausgewählt, d.h. solche Antworten, von denen ein Klassifikator besonders viel lernen kann. Wir identifizieren solche Antworten durch Uncertainty-Sampling-Methoden und erreichen dadurch mit einer gegebenen Anzahl von Annotationsschritten eine höhere Klassifikationsgenauigkeit als mit zufällig ausgewählten Antworten. In unserem zweiten Forschungszweig nutzen wir Clusteringmethoden um ähnliche Antworten zu gruppieren, so dass Gruppen von Antworten in einem Annotationsschritt bewertet werden können. Dadurch kann die Anzahl der insgesamt nötigen Bewertungsschritte drastisch reduziert werden.

Beim Vergleich zwischen clusteringbasierten Bewertungsverfahren und klassischem überwachten maschinellen Lernen, bei dem menschliche Annotationen dazu genutzt werden, einen Klassifikator zu trainieren, erbringen überwachte maschinelle Lernverfahren immer noch eine höhere Bewertungsgenauigkeit. Demgegenüber bringen Cluster den Vorteil eines strukturierten Outputs mit sich. Wir sind jedoch in der Lage, einen Teil diese Genauigkeitslücke zu schließen, in dem wir überwachte Featureauswahl und halbüberwachtes Clustering anwenden.

In einer zusätzlichen Studie untersuchen wir die automatische Verarbeitung von Lernersprache im Hinblick auf die Performanz von Werkzeugen für das Wortarten-Tagging. Wir annotieren ein deutsches Leseverstehenskorpus manuell sowohl mit Normalisierungsinformation in Bezug auf Rechtschreibung als auch mit Wortartinformation. Als Ergebnis der Studie finden wir, dass die Performanz bei der automatischen Wortartenzuweisung durch Rechtschreibkorrektur verbessert werden kann, insbesondere wenn wir den Lesetext als zusätzliche Evidenz dafür verwenden, welche Wörter der Leser in einer Antwort vermutlich benutzen wollte.

Erweiterte Zusammenfassung

Diese Arbeit beschäftigt sich mit Fragestellungen aus dem Bereich des automatischen Scorings von Short-Answer-Fragen. Short-Answer-Fragen sind ein weit verbreiteter Aufgabentyp in vielen Bildungsbereichen, bei denen Lerner eine Frage mit einem kurzen Text in natürlicher Sprache beantworten, dessen Länge sich zwischen wenigen Wörtern und mehreren Sätzen bewegt. Sie kommen als Lese- oder Hörverstehensaufgaben für Fremdsprachenlerner vor, finden aber beispielsweise auch als Wissens- oder Verständnisfragen in den Naturwissenschaften Anwendung. Short-Answer-Scoring grenzt sich vom verwandten Bereich des Essay-Scorings dadurch ab, dass beim Short-Answer-Scoring allein der Inhalt, aber nicht die linguistische Form der Antworten zur Bewertung herangezogen wird. Das bedeutet, dass ein Lehrer bei der Korrektur soweit wie möglich von Rechtschreib- oder Grammatikfehlern abstrahiert.

Als Beispiel für eine solche Aufgabe betrachten wir eine Aufgabe aus dem deutschen CREG-Corpus, der Leseverständnisaufgaben für Deutschlerner enthält. Gegeben ist der folgende (stark gekürzte) Text:

(0.1) *Als Frau Muschler auf dem Dachboden ihre Wäsche aufhing, kam die Nachbarin, die in ihrem Verschlag gekramt hatte. Ich habe etwas für Julchen zu Weihnachten, sagte sie. Wie nett von Ihnen, sagte Frau Muschler, da wird sich Julchen gewiss freuen. (...)*

Und die dazugehörige Frage:

(0.2) *Was machte Frau Muschler, als sie die Nachbarin auf dem Dachboden traf?*

Eine korrekte Antwort auf diese Frage wäre z.B.:

(0.3) *Sie hing ihre Wäsche auf.*

Das automatische Scoring von Short-Answer-Fragen wurde zu einem wichtigen Thema im Kontext des computergestützten Sprachenlernens. Die manuelle Korrektur von großen Antwortmengen, wie sie beispielsweise bei Einstufungstests anfallen, sind für den Bewerter äußerst zeitaufwendig. Automatische Methoden bergen hier ein großes Potenzial menschlichen Korrekturaufwand drastisch zu reduzieren. Darüber hinaus garantieren sie eine größere Konsistenz als menschliche Bewerter.

Im weiteren Kontext der automatischen Sprachverarbeitung (NLP) ist das automatische Scoring eine interessante und herausfordernde Aufgabenstellung mit Querbezügen zu verschiedenen anderen NLP-Aufgaben. Für einen menschlichen Bewerter ist es meistens eindeutig, ob ein Lerner eine Frage richtig beantwortet hat. Diese Entscheidung automatisch zu treffen ist ungleich anspruchsvoller. Das liegt daran, dass Antworten zu einer bestimmten Freitextaufgabe linguistisch auf viele verschiedene Arten realisiert werden können.

Richtige Antworten zu dem oben genannten Beispiel könnten unter anderem auch wie folgt aussehen:

(0.4) **LA1:** *Als Frau Muschler die Nachbarin auf dem Dachboden traf, hing sie ihre Wäsche auf.*
LA2: *Frau Muschler hing ihre Wäsche auf dem Dachboden auf*
LA3: *Sie machen die Wäsche.*
LA4: *Die Frau hat ihre Wasche in dem Dachboden aufgehangt.*

Durch diese Vielfalt an Antwortmöglichkeiten ist es unmöglich, die Korrektheit der Antwort durch einen direkten Abgleich mit einer Musterantwort zu bestimmen. Dadurch sind Short-Answer-Fragen deutlich anspruchsvoller automatisch zu bewerten als beispielsweise Multiple-Choice-Fragen. Anstatt zu überprüfen, ob eine Lernerantwort 1:1 der Musterantwort entspricht, stellt sich viel mehr die Frage, ob Lerner- und Musterantwort Paraphrasen voneinander sind. Dadurch hat Short-Answer-Scoring eine starke Verwandschaft mit der Aufgabe der Paraphrasenerkennung. Oft sind Lerner- und Musterantworten nicht komplett deckungsgleich, auch nicht bei korrekten Lernerantworten, zum Beispiel weil die Lernerantwort zusätzliche aber nicht notwendige Details enthält. Solche Bezüge werden im NLP-Bereich durch textuelles Entailment modelliert. Ein Text A entailt einen Text B, wenn ein Mensch, der Text A liest und als wahr annimmt, typischerweise davon ausgehen würde, dass auch B wahr ist. Dies ist z.B. der Fall für die beiden Sätze *Sie hing ihre Wäsche auf* und *Sie hing ihre Wäsche auf dem Dachboden auf*, wobei der zweite Satz den ersten entailt.

Eine weitere Herausforderung beim Scoring solcher Antworten liegt in der hohen orthographischen Variabilität und dem häufigen Auftreten von grammatischen Fehlern. Menschliche Bewerter sind angehalten, solche Fehler zu ignorieren. Dabei bilden sie eine sogenannte Zielhypothese darüber, was der Lerner vermutlich sagen wollte. LA3 hätte beispielsweise als mögliche Zielhypothese *Sie machte die Wäsche*. Bei der automatischen Bewertung muss man mit diesen Abweichungen ebenfalls umgehen, z.B. durch automatische Normalisierung oder die Verwendung von Features, die diese Abweichungen mitberücksichtigen.

Diese Doktorarbeit legt ihren Schwerpunkt auf zwei Aspekte im Zusammenhang mit Short-Answer Fragen und ihrer Bewertung: Zum einen betrachten wir ein Leseverständnisszenario, bei dem Deutschlerner Fragen zu Lesetexten beantworten, wie wir es im Beispiel oben gesehen haben. Dabei untersuchen wir insbesondere die verschiedenen Beziehungen, die es zwischen Lesetexten, Lernerantworten und vom Lehrer erstellten Musterantworten gibt. Zum anderen untersuchen wir, wie der menschliche Bewertungsaufwand durch vollautomatisches und computergestütztes Bewerten reduziert werden kann. Unter vollautomatischem Scoring verstehen wir eine Bewertung durch überwachte maschinelle Lernverfahren. Dabei wird aus einer Menge von

manuell bewerteten Antworten ein Modell gelernt, das dann auf neue ungesehene Antworten angewendet werden kann, ohne dass bei diesem Schritt noch menschliche Unterstützung notwendig wäre. Beim computergestütztem Bewerten handelt es sich um ein Szenario, in dem Bewerter bei der Bewertung unterstützt werden, z.B. indem ähnliche Antworten automatisch gruppiert werden. Das bedeutet, dass in beiden Fällen eine gewissen Anzahl Daten manuell annotiert werden muss. Im ersten Fall geschieht diese Annotation der Trainingsdaten jedoch zeitlich getrennt vor der automatischen Bewertung der Testdaten, während im zweiten Fall der Bewerter direkt in die Annotation der Testdaten involviert ist, ohne dass im Vorhinein ein Modell gelernt wird.

Zur Untersuchung des ersten Aspekts unternehmen wir eine Reihe von Korpusannotationsstudien, die sowohl die Beziehungen zwischen Lerner- und Musterantworten beleuchten, als auch die Beziehung zwischen diesen Antworten und dem Lesetext, auf den sie sich beziehen. Wenn Lerner Leseverstehensaufgaben beantworten, ist es nur natürlich, dass sie dabei explizite Anleihen aus dem Text machen und lexikalisches Material wiederverwenden.

In der ersten Annotationsstudie gehen wir der Frage nach, ob wir den Satz oder die Sätze, die zur Formulierung einer Antwort herangezogen wurden, im Text zuverlässig identifizieren können, insbesondere auch bei falschen Antworten. Wir annotieren dazu Sätze aus dem Lesetext, die vermutlich bei der Formulierung einer Antwort benutzt wurden, und machen die zu erwartende Beobachtung, dass die meisten korrekten Antworten (sowohl Lerner- als auch Musterantworten) problemlos mit bestimmten Textpassagen in Verbindung gebracht werden können. Inkorrekte Antworten haben ebenfalls oft eine Verbindung zu bestimmten Textpassagen, die aber häufig für die jeweilige Frage gar nicht relevant sind. Auf Grundlage dieser Erkenntnisse entwerfen wir ein neues Baseline-Bewertungsmodell, das für die Korrektheit einer Antwort nur in Betracht zieht, ob der Lerner die Antwort an der richtigen Stelle im Lesetext gesucht hat oder nicht.

Nachdem wir diese Verbindungen in den Text identifiziert haben, untersuchen wir die semantischen Beziehungen zwischen den Antworten und den annotierten Sätzen. Dazu annotieren wir die Relation zwischen Lerner- und Musterantworten sowie zwischen Texten und Antworten mit Entailment-Relationen. Dabei unterscheiden wir zwischen Paraphrasenrelationen, Entailment in beiden Richtung, aber auch einem partiellen Overlap, sowie verschiedenen Fällen von Nicht-Entailment wie Unvereinbarkeit der Sätze, oder Antworten, die in keinem Entailment zur Musterantwort stehen, weil sie die Frage gar nicht beantworten. Im Gegensatz zur in der Literatur weitverbreiteten Annahme, dass das Bewerten von Short-Answer Fragen und das Erkennen von Textual-Entailment zwischen Lerner und Musterantworten sich direkt entsprechen, finden wir in dieser Studie heraus, dass die beiden Aufgaben nur nahe verwandt aber nicht vollständig äquivalent sind. Korrekte Antworten entailen meistens, aber nicht immer, die Musteranwort

und auch den relevanten Satz aus dem ensprechenden Lesetext. Das bedeutet, eine korrekte Antwort enthält meistens zumindest die Information aus der Musterantwort. In einigen Fällen wird eine korrekte Lernernantwort jedoch von der Musterantwort entailt, d.h. der Lernerantwort fehlt zusätzliche oder spezifischere Information aus der Musterantwort, ohne dass sie dadurch falsch würde. Dies ist ein Hinweis darauf, wie Lehrer in ihren Bewertungen vorgehen, indem sie beispielsweise Musterantworten formulieren, die keine minimalen Antworten sind oder Lernernatworten akzeptieren, auch wenn sie nicht alle geforderten Details enthalten. Inkorrekte Antworten stehen meist in keiner Entailmentrelation mit der Musterantwort, haben aber oft zumindest teilweisen Overlap mit dem Text.

Diese nahe Verwandtschaft zwischen textuellem Entailment und Short-Answer Scoring erlaubt es uns, Goldstandard-Entailment-Information zu benutzen, um die Performanz beim automatischen Bewerten zu verbessern. Wir benutzen die annotierten Verbindungen zwischen Lesetexten und Antworten darüber hinaus auch in einem Scoringansatz, der auf statistischem Alignment basiert und dabei Methoden aus dem Bereich der maschinellen Übersetzung nutzt. Kernidee dabei ist das Konzept der vergleichbaren Corpora, d.h. von zwei Textsammlungen, von denen wir wissen, dass sie die gleiche Bedeutung haben, und in denen sich einzelne Sätze oder Abschnitt entsprechen (aligniert sind). Wie wir gesehen haben, haben korrekte Lernerantworten annähernd die gleiche Bedeutung wie die zugehörige Musterantwort. Darüber hinaus haben wir in unserer ersten Studie zu Lernerantworten Textsätze identifiziert, die eine ähnliche Bedeutung haben wie diese Antworten. Wir nutzen dieses Wissen zum Aufbau von vergleichbaren Corpora, aus denen wir statistische Alignments zwischen Wörtern lernen. Dabei nutzt das Verfahren aus, dass Wortpaare, die häufig in alignierten Sätzen vorkommen, vermutlich die gleiche Bedeutung haben. Aus diese Alignments extrahieren wir Features für das automatische Scoring und erreichen dabei eine Scoringgenauigkeit, die mit Ansätzen, die ein wissensbasiertes Alignment nutzen, vergleichbar ist.

Im zweiten Teil der Arbeit beschäftigen wir uns damit, wie der Bewertungsaufwand beim Menschen verringert werden kann, wenn Antworten vom Lehrer manuell bewertet werden. Unsere Untersuchungen dazu basieren auf zwei Methoden: Active Learning und Clustering.

Beim Active-Learning-Ansatz wählt man aus der Gesamtheit der zur Verfügung stehenden Trainingsdaten vorrangig besonders informative Antworten zur Bewertung aus, d.h. solche Antworten, von denen ein Klassifikator besonders viel lernen kann. Wir identifizieren solche Antworten durch Uncertainty-Sampling-Methoden. Dabei wird iterativ im Scoring-Prozess immer wieder die Antwort bestimmt, bei deren Klassenzugehörigkeit der Klassifikator sich besonders unsicher ist, z.B. weil sie sich auf der Entscheidungsgrenze zwischen zwei Klassen befindet. Wir vergleichen diese Verfahren mit einer mehrfachen zufälligen Auswahl von Trainingsdaten und erreichen durch Uncertainty-Sampling bei einer gegebenen Anzahl von Annotationsschritten

eine deutlich höhere Klassifikationsgenauigkeit als mit zufällig ausgewählten Antworten.

In unserem zweiten Forschungszweig nutzen wir Clusteringmethoden. Dabei bauen wir auf der Beobachtung aus, dass zueinander sehr ähnliche Antworten oft entweder alle richtig oder alle falsch sind. Darüber hinaus entsprechen Gruppen von ähnlichen falschen Antworten oft auch einer gemeinsamen falschen Idee von der Antwort. Wenn ähnliche Antworten gruppiert werden, hat das für den Bewerter den Vorteil, dass Gruppen von Antworten in einem Annotationsschritt bewertet werden können und darüber hinaus auch die Möglichkeit besteht, Feedback zu ganzen Clustern von Antworten zu geben. Wir nutzen bei der Bewertung von Clustern das Verfahren der Label-Propagation, bei dem wir simulieren, dass der Bewerter nur ein Item pro Cluster bewertet und diese Bewertung dann auf alle Antworten aus dem Cluster übertragen wird. Wir wählen dazu ein prototypisches Item aus dem Zentrum des Clusters aus und erreichen bessere Ergebnisse als durch eine zufällige Auswahl eines Items für die Label-Propagation. Auf diese Weise kann in einem manuellen Scoring-Szenario auf Hörverstehensdaten die Anzahl der insgesamt nötigen Bewertungsschritte drastisch reduziert werden, ohne dass der Lehrer die Kontrolle über den Scoringprozess komplett an ein automatisiertes Verfahren abgibt.

In einer zweiten Studie vergleichen wir clusteringbasierte Bewertungsverfahren mit klassischem überwachten maschinellen Lernen, bei dem menschliche Annotationen dazu genutzt werden, einen Klassifikator zu trainieren. Dabei erbringen überwachte maschinelle Lernverfahren immer noch eine höhere Bewertungsgenauigkeit. Demgegenüber bringen Cluster jedoch den Vorteil eines strukturierten Outputs mit sich, bei dem der Bewerter einen besseren Überblick über das Spektrum der abgegebenen Antworten erhält. Indem wir überwachte Featureauswahl und halbüberwachtes Clustering anwenden, sind wir in der Lage, einen Teil der Genauigkeitslücke zu den überwachten Lernverfahren zu schließen. Es zeichnet sich ab, dass sich Clusteringverfahren insbesondere für kürzere Antworten anbieten und beispielsweise im Bereich des Hörverstehens dazu geeignet sind, Antworten mit einer hohen Rechtschreibvariabilität in Gruppen zusammenzufassen.

In einer zusätzlichen Studie wenden wir uns der oben angesprochenen Abweichung von der Norm zu, die wir bei Lernersprache beobachten können. Dazu untersuchen wir die automatische Verarbeitung von Lernersprache im Hinblick auf die Performanz von Werkzeugen für die automatische Zuweisung von Wortarten-Tags. Wir annotieren das CREG-Corpus manuell sowohl mit Normalisierungsinformation in Bezug auf Rechtschreibung als auch mit Wortartinformation. Als Ergebnis der Studie finden wir heraus, dass die Performanz bei der automatischen Wortartenzuweisung durch Rechtschreibkorrektur verbessert werden kann, insbesondere wenn wir den Lesetext als zusätzliche Evidenz dafür verwenden, welche Wörter der Leser in einer Antwort vermutlich benutzen wollte.

Zusammenfassend haben wir in dieser Arbeit mehr über das Wesen von Short-Answer-Fragen

gelernt, sowohl im Hinblick darauf, wie Lerner Leseverstehensfragen beantworten, als auch, wie Lehrer diese Antworten bewerten. Während vorherige Arbeiten Lernerantworten im Wesentlichen mit der Musterantwort verglichen haben, haben wir in dieser Arbeit auch den Lesetext mit in Betracht gezogen.

Darüber hinaus haben wir Wege aufgezeigt, wie der menschliche Bewertungsaufwand reduziert werden kann, indem nur besonders relevante oder prototypische Datenpunkte bewertet werden und dies in Active-Learning und Clustering-Experimenten evaluiert. Diese Experimente haben den Weg bereitet für praxisorientiertere Studien, die automatisches Scoring in den tatsächlichen Lehralltag integrieren sollen und die es uns schlieülich ermöglichen werden, den Nutzen der vorgestellten Methoden in der Praxis zu bestätigen.

Contents

List of Figures

List of Tables

1 Introduction

This thesis addresses the topic of automatic assessment in an educational setting at the example of automatic short-answer scoring. We analyze two aspects of short-answer questions and their scoring. First, we focus on a reading comprehension scenario, which consists of a reading text with a question about that text, and learner answers, as well as a teacher-specified target answer given in response. We investigate the mutual relations these texts stand in. Second, we determine ways how to reduce human scoring workload by automatic and assisted scoring.

The thesis is situated both in the context of *computer-assisted language learning* (CALL) and *educational technologies*, two fields with growing importance over recent years. On-line exercises, tests and whole on-line courses have become more and more popular. People use on-line platforms such as Duolingo[1] or Babble[2] (both founded in the late 2000s) to learn a foreign language at home in their free-time, schools and universities supplement traditional classes for a variety of subjects with on-line activities using platform such as the 2002 released Moodle system[3] or offer MOOCS – massive open on-line courses –, and providers of language courses conduct placement tests on-line even before the prospective student arrives on location. Within the NLP community, growing interest in CALL and educational technologies shows in rising submission numbers to workshops such as the BEA series established in 2003.

Computer-assisted learning offerings have the huge advantage that they scale to an arbitrary number of people at any location, where traditional classes can usually cater only for a limited number of students physically present in class. Students can participate in on-line learning largely in the absence of a human teacher. Of course, teachers or other experts have the role of content creators and are often also available via email or forums to answer particular questions, but many activities take place without direct human supervision.

Automatic assessment is an important part in almost any on-line learning or testing scenario. It is a type of *evaluation mechanism* that either acts as a feedback device and reports back to a learner whether they solved a task correctly (*formative feedback*), or informs the teacher or test evaluator about the student's performance (*summative feedback*). In general, such an evaluation mechanism can be instantiated in various ways that we will briefly discuss here; by a human,

[1] https://www.duolingo.com/
[2] https://www.babbel.com/
[3] https://moodle.org/

fully automatically or semi-automatically .

Human evaluation is the oldest and most natural form of assessment and can take place by means of self-evaluation, peer evaluation or expert evaluation: In the simplest form, the learner is shown the correct solution and asked to judge their own answer in comparison (self-evaluation). In peer evaluation, feedback is given by fellow learners. This procedure is, for example, implemented in on-line systems by some language learning platforms like busuu[4] or lang-8[5], where writings by learners of a certain language are corrected by other learners who are native speakers of that language. The traditional and dominant form of evaluation, especially in classroom-based settings, is expert evaluation, where a teacher gives feedback. That typically means that a teacher has to assess each learner answer individually, a time-consuming task that makes it desirable to automate this process.

When evaluation is done *fully automatically*, no direct human intervention is necessary (see Chapter 5). This means that the learner may receive immediate feedback, such as information on whether the answer they gave was correct or incorrect, and a teacher might get test results instantaneously. For exercise types like multiple choice questions, automatic evaluation is obviously trivial. In general, fully automatic evaluation can be done easily when the number of correct answers is limited and they can be enumerated preemptively by a teacher.

This is typically not the case for *free-form answers in natural language*, which we consider in this thesis. Free-form answers are very common in many educational scenarios. They offer the advantage that a student has to formulate an answer on their own, thereby showing that they understood the material, instead of, for example, just selecting one solution out of several alternatives as is done in multiple choice exercises. For almost all exercises, the most natural way to give an answer would be in natural language (with the exception of, e.g., mathematical exercises asking for a proof or a computation, or questions, where a drawing or a diagram would be part of a good answer). However, the automatic processing of natural language answers suffers from the problem of language variation: there is typically an unbounded number of ways of expressing the same idea. Enumerating all correct solutions is often possible for very restricted tasks like filling a blank with an appropriate grammatical form, but language variation makes listing all possible correct solutions infeasible for many tasks where the *content* of an answer is to be evaluated.

Short-answer questions, as the example below, are a prominent type of such a task. They occur frequently in contexts such as reading and listening comprehension for foreign language learners or science questions for high school students. They ask for answers that typically have a length between just one phrase and a few sentences, thus distinguishing them from longer essays. They

[4] www.busuu.com
[5] http://lang-8.com

are evaluated solely based on their content while teachers try to ignore grammatical or spelling errors.

Consider the following example from the German reading comprehension data set CREG, one of the data sets used for this thesis, consisting of a question about a text, the *target answer* specified by a teacher and a number of correct learner answers, which are all different from each other despite expressing the same content:

(1.1) **Question:** Was machte Frau Muschler, als sie die Nachbarin auf dem Dachboden traf?

What was Ms Muschler doing when she met her neighbor in the attic?

Target Answer: Sie hing ihre Wäsche auf.

She was hanging up the laundry.

Learner Answers (all correct):

LA1: Als Frau Muschler die Nachbarin auf dem Dachboden traf, hing sie ihre Wäsche auf.

When Ms Muschler met her enighbour in the attic, she was hanging up the laundry.

LA2: Frau Muschler hing ihre Wäsche auf dem Dachboden auf.

Ms Muschler was hanging up her laundry in the attic.

LA3: Sie machen die Wäsche.

She do the laundry.

LA4: Die Frau hat ihre Wasche in dem Dachboden aufgehangt.

The woman was hanging up her laundry within the attic.

CREG is one out of four data sets used in this thesis. Apart from CREG, we use ASAP, an English short-answer data set collected from US high school students covering 10 prompts from various domains, the Powergrading data set with prompts from US immigration tests and the Laempel data set, containing listening comprehension data collected at Saarland University from placement tests for learners of German as a foreign language. For a detailed description of the data see Chapter 3.

The automatic evaluation of the semantic content of a free-form answer is an important task from an educational perspective, because human scoring can be a time-consuming and tedious task.

It is also a challenging task from an NLP perspective. Short-answer questions have become popular for automatic evaluation because of their intermediate linguistic complexity: they have a complexity that goes beyond filling a gap with a single word both in terms of answer length

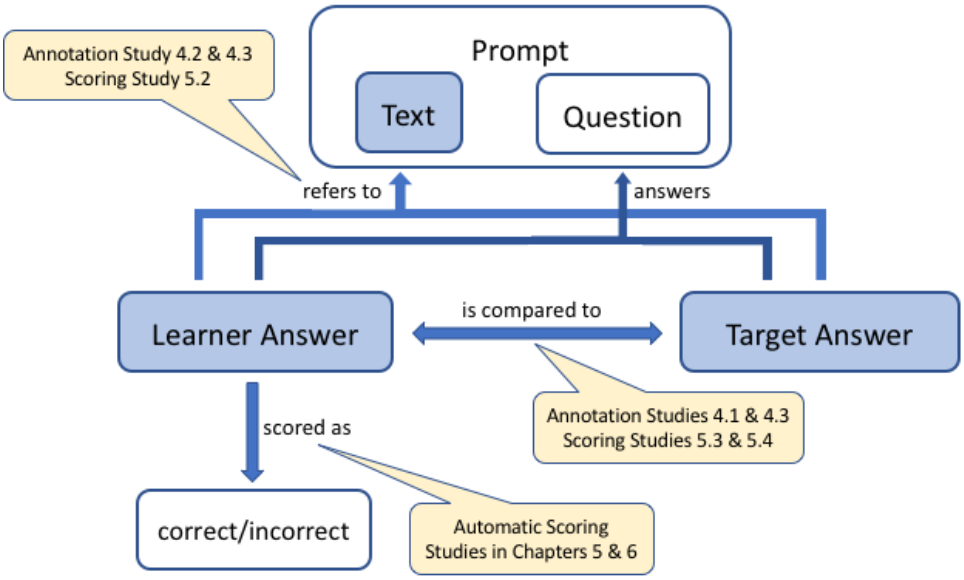

Figure 1.1: Main components of the short-answer question scenario. Interactions between components are indicated by arrows. We also provide pointers how the work in this thesis links to these interactions

and language variance in the answer, but is limited compared to longer pieces of writing.

We investigate in this thesis an additional way to take a part of the scoring burden from a teacher by *semi-automatic evaluation* (Chapter 6). In this type of assessment, the teacher is still involved in assessing learner answers, but receives substantial help from an automated system. Such a system is implemented in this thesis by grouping similar answers together so that the whole group can be scored in one step instead of scoring each answer individually. Such semi-automatic scoring comes in handy when a teacher has to grade a large number of answers to the same question, as is, e.g., often required when conducting placement tests.

Figure 1.1 gives an overview of the main components of a short-answer question and their interactions. Note that the reading text in the prompt is optional. It occurs, e.g., in reading comprehension, but might be absent in science questions. The figure shows how the work in this thesis hooks into the different components and interactions.

Lessons learned from related NLP tasks can be leveraged to work on the problem of evaluating short-answer question (see Section 2.2). When assessing the correctness of an answer to a short-answer question, one can often rely on a teacher-specified sample solution, the target answer, and compare the learner answer to it. Ideally, a correct learner answer has the same semantic content

as the target answer, casting the ASAS problem as one of *paraphrase detection* or *detecting semantic similarity*. We build on this notion in our content scoring approaches that rely on statistical alignments between target and learner answers (Section 5.4), as well as in clustering approaches, which aim at building clusters of highly similar answers (Section 6.2 and 6.3).

However, the binary distinction whether a learner answer is a paraphrase of the target answer is not enough. A learner answer containing additional, but unnecessary content is often still considered as correct by teachers in the classroom; such an answer textually entails the target answer, but not vice versa. An answer containing less information than the target answer, i.e., an answer that is entailed by the target answer, is often considered incorrect. Therefore, ASAS is similar to and has also often been equated with *recognizing textual entailment (RTE)*. Textual entailment is a binary relation between two texts; it is given if people reading one text would typically infer that the other text is most likely also true (Dagan et al., 2013). We annotate such entailment relations in Chapter 4 and use them for automatic scoring in Section 5.3.

ASAS also has connections to *question answering*. The task of question answering is to find the answer to a question based on some text corpus. Analogously, in reading or listening comprehension exercises, a learner has to find the answer to a question on the basis of a reading or listening text. Scoring such an answer in ASAS corresponds to evaluating the appropriateness of a suggested answer in question answering.

An additional challenge when dealing with texts written by learners are the particular properties of the language used in such answers. For corpora from the area of language learning, this language is *learner language*, i.e., the imperfect language variety of non-native speakers, for corpora from other educational domains, it is language written by native, but *non-professional writers*; we will call this variant *student language*. Both these language varieties contain all sorts of spelling errors as well as grammar problems. While the explicit recognition or correction of such deviations from standard language is not the job of ASAS (but rather that of writing assistant programs), interpreting answers to short-answer questions still requires dealing with noisy user-input, which makes both their automatic processing and also the scoring by humans harder. Running NLP processing tools on the original learner and student data typically leads to a decreased performance as compared to standard texts. When teachers have to deal with learner language, they typically form a so-called *target hypothesis*, a standard language version of what the learner presumably wanted to say. This is known to be difficult for humans. In automatic processing, the problem of learner language can be tackled in a similar way by normalizing the data before processing. Alternatively, linguistic processing tools can be adapted to the properties of learner and student language. For both of these tasks we can relate to solutions for other kinds of non-standard language such as data from computer-mediated communication. We investigate the effects of normalization and tagger adaptation on POS tagging in Chapter 7.

In the following, we will specify the research questions addressed in this thesis and the answers we provide.

1.1 Research Questions and Contributions of this Thesis

In this thesis, we address the general topic of automatic evaluation in an educational setting. We consider the specific task of scoring short-answer questions because of the linguistic complexity of the expected answers that makes the task challenging but not utterly unfeasible. We investigate the linguistic nature of short-answer questions and the implications this has on automatic scoring, as well as the amount of human scoring work needed for automatic and semi-automatic scoring.

More precisely, we investigate the following interrelated sub-areas of ASAS in the thesis:

1. In a series of corpus studies, we have a closer look at the properties of answers to short-answer questions (SAQs) for reading comprehension in terms of their semantic relations to their corresponding target answers and reading texts (see Chapter 4).

2. We investigate different kinds of automatic scoring models for ASAS. We present ways in which findings from the first part, such as strong links between learner answers and the corresponding reading text, can be used in grading (Chapter 5).

3. We then present studies that explore how human annotation effort can be reduced for scoring by means of clustering and active learning (Chapter 6).

4. To complement our studies that all work with learner language, we present an additional study which focuses on the linguistic preprocessing of learner language material (Chapter 7).

First, we investigate the nature of learner answers given in a reading comprehension scenario. In reading comprehension, a student reads some text, e.g., a newspaper article, and has to answer questions about it. We want to address the semantic relationships between learner answers and both the connected reading text and the corresponding target answer, as well as between the reading text and the target answer. This leads to our first research question:

RQ 1.1: How can the reading text be used in ASAS for reading comprehension tasks?

Many approaches for ASAS rely on similarity between the learner answer and a target answer. In a reading comprehension scenario, obviously, many language learners copy individual words, phrases or even whole clauses from the text when formulating their answers to a question. This

learner strategy is known as *lifting*. Therefore, we propose to use also the reading text as a valuable source of information for automatic evaluation.

We show through an annotation study on about 1000 learner answers for the German reading comprehension corpus CREG that a learner answer from the CREG corpus can most of the time be linked reliably by human annotators to one or a few specific text sentences where the information originates from (often verbatim). The target answer can be linked to the text in this way, as well (see Section 4.2). We further show that we can reliably determine this best sentence automatically using alignment methods originally proposed for comparing learner answers to target answers by applying them to answers and text sentences. We find that such information is sufficient for a simple rule-based classifier that can be used as a baseline for automatic scoring while being far from beating the state of the art for scoring algorithms on this data set. We further investigate how to enrich an alignment-based scoring system proposed in the literature with text-based features and find a tendency for improvement (Section 5.2).

RQ 1.2: How closely are the two tasks of RTE and ASAS related?

The ASAS task has been compared and equated to the problem of recognizing textual entailment (RTE): A learner answer is said to be correct if it is at least as specific as the target answer, i.e., it entails the target answer, and as incorrect otherwise (Section 2.2.1).

We test this claim for the CREG corpus by means of an annotation study that investigates the entailment relation between learner answers and their corresponding target answers (Section 4.1). We find that the claim mostly holds. However, there is the problematic field of reverse entailment (target answer entails learner answer) and partial overlap, for which the hypothesis predicts an incorrect answer, while the majority of these answers is actually labeled as correct. We conclude that target answers in the CREG corpus cannot be seen as minimal correct solutions containing only absolutely necessary input, but rather as suggestions for potential correct answers, and that teachers are often lenient when scoring answers containing less detail than expected.

In a second annotation study, we address the entailment relation between learner answers and the corresponding text (Section 4.3). In a reading comprehension scenario we would intuitively assume and also find in our study that a correct learner answer has to be supported by the text. For an incorrect answer, there are cases where the answer is equally supported by the text, but irrelevant to the question as well as cases where the answer is not fully supported by the text. We show in oracle experiments that gold-standard entailment information contributes to grading accuracy when used as a feature and that a standard feature set for automatic scoring can also be used to predict entailment relations between learner and target answers (Section 5.3).

We then address various ways to reduce human annotation effort in an ASAS scenario by means of computational models. We investigate two general types of models: (a) *prompt-independent models* are learned and applied across prompts, while (b) for *prompt-specific models* one model per prompt is learned. Prompt-independent models already reduce human annotation effort by generalizing over individual prompts: after one model is learned from a number of different prompts and answers, it can be applied to either new answers from the same prompts or to answers to new prompts of a similar kind. This type of model has been applied in the literature to SAQs eliciting relatively short answers that are compared to a target answer. The correctness decision is based on the similarity between a learner answer and the corresponding target answer, for example by comparing them on the semantic or on the surface level.

Prompt-specific models can only be applied if there is enough training material per prompt. They have the advantage that they make use of lexical features such as lemma n-grams or dependency triples specific to a certain prompt. Instead of learning the contribution of different similarity measures between a learner answer and a target answer to the correctness of an answer, they learn the contribution of the presence and absence of a particular n-gram such as *"lives in Berlin"*. Such a behavior is especially beneficial in scenarios where answers are long and therefore have high lexical variety (such that a correct answer does not necessarily have a high similarity with a target answer). It is crucial for scenarios where target answers are not available at all.

Prompt-independent approaches to ASAS often rely on some sort of alignment between a learner answer and its target answer. Such alignments are traditionally knowledge-based, i.e., they explicitly specify which types of correspondences between linguistic units may trigger an alignment. We propose an alternative approach inspired by statistical machine translation (SMT) and assume that alignments can be learned from the data directly.

RQ 2.1: Can statistical alignment for prompt-independent scoring be used as an alternative to knowledge-based alignment?

In SMT, one uses sentence-aligned parallel corpora in two languages to find phrases that frequently co-occur. If, for example, most of the English sentences in a corpus containing the phrase *"the black dog"* have German translational counterparts containing the phrase *"der schwarze Hund"*, we assume that they mean the same in both languages. We apply this approach in the ASAS scenario and create a comparable corpus of learner and target answers for which we perform statistical alignment. Our method is able to extract meaningful paraphrases, and

using properties of the alignments as features, we can reach a classification performance that is on level with knowledge-based alignment approaches on the CREG corpus.

Alternatively, we address a class of models where a classifier per prompt is learnt. If such a kind of model is trained for a new prompt, new training material has to be annotated (assuming that supervised machine learning is used to train the model). Therefore, it is crucial for this type of ASAS approach to keep the number of training instances small. We address this problem in two ways, through active learning and clustering. For these experiments, we switch from the CREG corpus with only few learner answers per question to data sets that contain substantially more instances per prompt.

RQ 2.2: Are active learning methods suitable in a prompt-specific ASAS scenario?

The key idea in active learning is to select training instances for a supervised machine learner iteratively in such a way that they are more informative than a baseline of randomly selected training instances. We show for the English ASAP corpus that uncertainty-based active learning methods have the potential to reduce human annotation workload, but observe at the same time a high variability across prompts that we can partially explain by class imbalance and class separability.

RQ 2.3: Can clustering reduce a teacher's workload in an assisted scoring scenario?

Our core assumption is that similar answers to the same prompt are likely to receive similar grades and can therefore be graded together by a teacher in one scoring step instead of scoring each item individually. Therefore we cluster learner answers and introduce the concept of label propagation, where a teacher only scores one or a few instances out of each cluster, whose label is then assigned to all cluster members.

In a first set of experiments on the Laempel data of listening comprehension data, we show that a teacher only needs to score 40% of all items in order to reach a grading accuracy of 90% when labeling and propagating cluster centroids. This is a substantial reduction in comparison to traditional pen-and-paper grading.

In a second set of experiments on the ASAP data, we compare clustering to supervised machine learning, with the idea that human annotation effort could either be used in clustering or in the training of a supervised machine learner. In addition to labeling cluster centroids resulting from unsupervised clustering, we propose to reserve some of the human annotation effort to label instances before clustering and use these items for feature selection and as constraints in a semi-supervised clustering approach. In doing so, we can further improve clustering performance

without increasing the overall human scoring work load. We come closer to the performance of supervised machine learning.

Finally, we address the automatic linguistic processing of learner language. While this topic is not directly linked to automatic scoring, we consider it a useful add-on which gives us more insights into the particularities of learner language in short-answer data and what the effects on linguistic processing compared to standard newspaper text are.

RQ 3.1: How can we improve linguistic preprocessing of learner answer data?

Many ASAS approaches build on linguistic preprocessing, such as POS tagging. However, such tools are usually trained on newspaper texts and thus performance decreases when they are applied to other domains. We explore two methods commonly used in domain adaptation tasks: Normalization of learner answers before processing and adaptation of POS taggers. We exploit the nature of the reading comprehension task and the lifting behavior of students for both of these methods in our treatment of words that are out-of-vocabulary for a POS tagger. We identify words that are misspellings of prompt material and normalize them accordingly and add unknown words to the tagger lexicon whenever we can assume that they are not misspellings because they occur in the reading text. To evaluate our method we annotate normalization and POS information for the CREG corpus and show that our method improves POS tagging for out-of-vocabulary words substantially.

1.2 Structure of this Thesis

In Chapter 2 of this thesis, we will introduce the ASAS task in more detail: We will see what short-answer questions are, how they are used, and why automatic scoring of such tasks is both challenging and important (Section 2.1). We will relate ASAS to other NLP tasks in Section 2.2 and give an overview of previous short-answer scoring approaches and how they fit into our distinction of models across prompts and models per prompt in Section 2.4. In Chapter 3, we present the data sets that are used in our studies and review data sets used in general for the task of ASAS.

Chapters 4 to 7 present the main studies and experiments of this thesis as described in the previous section: In Chapter 4, we present corpus studies about the semantic relation between learner answers and target answers and answers and prompts; Chapters 5 and 6 present our work on ASAS investigating both prompt-independent models and the reduction of human annotation effort for prompt-specific models. In Chapter 7, we present experiments about the preprocessing of learner language. Chapters 8 and 9 provide conclusions and an outlook to future work.

1.3 Publications

The thesis includes work first published in the following papers:

- Andrea Horbach, Alexis Palmer, and Manfred Pinkal: *Using the text to evaluate short answers for reading comprehension exercises*. *SEM 2013.
- Andrea Horbach, Alexis Palmer, and Magdalena Wolska: *Finding a Tradeoff between Accuracy and Rater's Workload in Grading Clustered Short Answers.* LREC 2014.
- Andrea Horbach and Alexis Palmer: *Investigating Active Learning for Short-Answer Scoring.* BEA 2016 workshop.
- Andrea Horbach and Manfred Pinkal. *Semi-supervised Clustering for Short-Answer Scoring.* LREC 2018.

Additionally, the following publications are extensions of Bachelor and Master theses supervised by the author; in each case, the author has made significant contributions to the published papers:

- Simon Ostermann, Andrea Horbach, and Manfred Pinkal: *Annotating Entailment Relations for Shortanswer Questions.* NLP-TEA 2015.
- Nikolina Koleva, Andrea Horbach, Alexis Palmer, Simon Ostermann, and Manfred Pinkal: *Paraphrase Detection for Short Answer Scoring.* 3rd NLP4CALL Workshop, 2015.
- Lena Keiper, Andrea Horbach, and Stefan Thater: *Improving POS Tagging of German Learner Language in a Reading Comprehension Scenario.* LREC 2016.

2 Background and Related Work

Automatic short-answer scoring (ASAS) deals with the automation of a task that comes from the educational domain, namely that of scoring answers to short-answer questions, by means of natural language processing (NLP). This chapter provides both the educational background and the NLP background for the task and, in addition, presents related work in ASAS. We specify the task of ASAS in Section 2.1; we compare ASAS to other NLP tasks in Section 2.2; in Section 2.3, we discuss the specific problems that arise from dealing with learner language; finally, we present previous approaches to ASAS in Section 2.4.

2.1 The Task of Automatic Short-Answer Scoring

In this section, we introduce the ASAS task. We will (1) discuss what short-answer questions are, and for what kinds of applications they are used, (2) discuss why the automatic scoring of answers to such questions is both important and challenging, and (3) for which types of short-answer questions ASAS is applicable.

2.1.1 Short-Answer Questions and their Applications in Educational Contexts

ASAS is the task of automatically assigning a label or score to an answer given in response to a short-answer question (SAQ). An SAQ, as for example defined in Burrows et al. (2014), is an open-ended question, i.e., a question that asks for (typically written) free-form input in natural language. The length of answers to short-answer questions ranges from a single phrase to a few sentences; questions asking for longer responses are called *essay questions*. When evaluating answers to SAQs, only the *semantic content* of the answer is important; writing style or the linguistic correctness of an answer, i.e., the question of whether it contains spelling or grammatical errors is not taken into consideration. SAQs are often used to ask for factoid knowledge, but might also, for example, ask for personal opinions, especially when used in a traditional classroom setting. (We will see in Section 2.1.3 that this second type of questions is problematic for automatic scoring.)

TEXT: SCHLOSS PILLNITZ
Das Schloss, das im Osten Dresdens liegt, ist für mich das schönste Schloss in Dresdens Umgebung. (...) Eine besondere Attraktion im Park ist die Kamelie. Die mittlerweile über 230 Jahre alte und 8,90 m hohe Kamelie bekam 1992 ein fahrbares Haus, in dem Temperatur, Belüftung, Luftfeuchte und Beschattung durch einen Klimacomputer geregelt werden. In der warmen Jahreszeit wird das Haus neben die Kamelie gerollt. Während der Blütezeit von Mitte Februar bis April trägt sie zehntausende karminrote Blüten. Ableger der Pillnitzer Kamelie werden jedes Jahr in begrenzter Zahl während der Blütezeit verkauft, dann ist ein Besuch besonders lohnend.

This palace, which lies in the east of Dresden, is to me the most beautiful palace in the Dresden area. (...) One special attraction in the park is the camellia tree. In 1992, the camellia, which is more than 230 years old and 8.90 meters tall, got a new, movable home, in which temperature, ventilation, humidity, and shade are controlled by a climate regulation computer. In the warm seasons, the house is rolled away from the tree. During the Blossom Time, from the middle of February until April, the camellia has tens of thousands of crimson red blossoms. Every year, a limited number of shoots from the Pillnitz camellia are sold during the Blossom Time, making it an especially worthwhile time to visit.

QUESTION:
Ein Freund von dir möchte sich die alte Kamelienpflanze ansehen. Wann sollte er nach Pillnitz gehen und warum gerade in dieser Zeit?
A friend of yours would like to see the historic camellia tree. When should he go to Pillnitz, and why exactly at this time?

Figure 2.1: Example of a reading comprehension task (CREG)

In educational contexts, SAQs are used in a variety of scenarios and can be considered a standard exercise type. In second language acquisition, SAQs are one way of testing reading and listening comprehension, i.e., they are used for the assessment of a learner's receptive written and oral skills.[1] Reading or listening comprehension questions are designed to test whether a learner understood a text that they have read or listened to. Figure 2.1 contains an example for a reading comprehension task, where a student reads a text and has to answer a question about it. (All corpora from which this and the following examples are cited are presented in detail in Chapter 3.)

SAQs are also used in reading comprehension tasks targeted at native speaking students, although the nature of the questions typically differs. SAQs for second language learners, especially for beginners, often address basic factoid knowledge from a text, as we see in the example above, taken from a corpus of reading comprehension questions for learners of German as a foreign language. In contrast, reading comprehension for native speakers, e.g., high school students, typically asks for answers that involve a higher degree of reasoning and abstraction from the text. One such corpus, which is not specifically targeted at language learners, for example, contains reading comprehension exercises that present a longer story and then ask a question

[1] Writing and speaking are the corresponding productive skills according to the division of language learning into four skills that has become a quasi-standard among language teachers (Hinkel, 2010).

like the following:

(2.1) Identify ONE trait that can describe Rose based on her conversations with Anna or Aunt Kolab. Include ONE detail from the story that supports your answer. [ASAP]

In such kinds of questions, a student has to infer or compose a correct answer from a longer part of the text, instead of identifying a single piece of information.

Apart from reading comprehension exercises, SAQs are also used to test factoid knowledge for native speakers, for example in the sciences. One example is the question from a corpus containing computer science questions:

(2.2) What is the role of a prototype program in problem solving? [Mohler & Mihalcea]

Another application for short-answer questions is testing of analytical and problem solving capabilities, as, e.g., in the following question:

(2.3) After reading the groups procedure, describe what additional information you would need in order to replicate the experiment. Make sure to include at least three pieces of information. [ASAP]

Such questions often come with additional material, as in the case of the example question above a description of an experiment and an outcome table, but also in the form of graphs or statistics to be evaluated.

Short-answer questions can be used for two different assessment scenarios, *summative* and *formative* assessment (Scriven, 1967). Formative assessment should support the learning process. It is typically used in low-stake tests, such as homework assignments, and is meant to provide feedback to students how to improve their answers. *Summative assessment*, in contrast, is meant to inform a teacher about the proficiency of their students, i.e., it is used for grading. It often occurs in high-stakes tests, such as exams. Different types of scores are associated with these assessments: summative feedback frequently consists of a numeric score or binary correctness label that can easily be added up to an overall score for an exam and can then be mapped to a grade by the teacher. Formative assessment usually provides a more informative feedback message to the student. A teacher might give a student feedback such as "Your answer is correct, but you are missing important details." We will see in the next section, how these two types of assessment reflect in different label sets for automatic scoring.

In summary, SAQs are used in a variety of different educational scenarios and subjects, highlighting their importance for teaching. After this presentation of the general usage of SAQs, we will next discuss automatic grading of such questions.

2.1.2 Short-Answer Questions and Automatic Scoring

SAQs have traditionally been applied in classroom-based teaching and testing scenarios, either in class, as homework or as part of tests. They are usually graded by a teacher, who marks answers as correct or incorrect or assigns a number of points and potentially also gives additional feedback to the student. In automatic short-answer scoring the role of the teacher is (partially) automated: instead of a human grading each answer individually, the answers are graded by an automatic scoring mechanism. Data which has been manually scored by a teacher serves as the gold-standard for automatic scoring, i.e., automatic scoring methods try to model a teacher's scoring behavior as closely as possible. We will give an overview of methods used in ASAS in Section 2.4.

Automatic evaluation and scoring in an educational context has become more and more important with the advent of on-line language courses and on-line learning platforms such as Moodle[2]. In many such applications, automatic evaluation is restricted to basic exercise types such as multiple choice, re-ordering or simple gap-filling questions, and most free-text answers have to be scored manually by a teacher. Automatic scoring is trivial only in cases where the number of correct solutions is limited and all possibly correct solutions can be enumerated by a teacher, as is frequently, but not always, the case for instantiations of the aforementioned basic exercise types: a gap-filling exercise asking to add the correct determiners in a sentence can be scored automatically, because the number of correct solutions is very restricted. In contrast, an exercise asking to fill in an appropriate adjective in a sentence with a gap might have correct solutions unforeseen by a teacher and is therefore harder to automate.

SAQs, in contrast, have a linguistic complexity that typically leads to many (even arbitrarily many) correct answers for a single prompt. There are two reasons for this. First, for some questions there is more than one possible conceptually correct answer as shown in the following example

(2.4) Name one state [in the US] that borders Mexico. [Powergrading]

with *"California"*, *"Arizona"*, *"New Mexico"* and *"Texas"* as correct answers. In such cases, the number of different correct solutions is usually limited and would not be a challenge for automatic evaluation. But second, and crucially, there are most of the time many ways of phrasing the same answer in different words. Consider the following example, which we have already seen in the introduction, consisting of a question with a corresponding sample solution, the *target answer*, and a number of correct learner answers:

[2] https://moodle.org

(2.5) **Question:** Was machte Frau Muschler, als sie die Nachbarin auf dem Dachboden traf?
What was Ms Muschler doing when she met her neighbor in the attic?
Target Answer: Sie hing ihre Wäsche auf.
She was hanging up the laundry.

Learner Answers (all correct):
LA1: Als Frau Muschler die Nachbarin auf dem Dachboden traf, hing sie ihre Wäsche auf.
When Ms Muschler met her enighbour in the attic, she was hanging up the laundry
LA2: Frau Muschler hing ihre Wäsche auf dem Dachboden auf.
Ms Muschler was hanging up her laundry in the attic.
LA3: Sie machen die Wäsche.
She do the laundry.
LA4: Die Frau hat ihre Wasche in dem Dachboden aufgehangt.
The woman was hanging up her laundry within the attic
[CREG]

All of these answers are scored as correct by teachers, are different from each other, but express the same meaning. Paraphrasing the same content in different ways leads to this variety of answers. In addition, other linguistic phenomena like the repetition of question material, referencing to entities in the question via pronouns, as well as spelling variability and grammatical errors increase the number of different answers to a question further. We could easily imagine many more ways to phrase this sentence, such that it is impossible for a teacher to write down all variants of a correct answer.

Therefore, there is typically a large number of unique answers in ASAS data sets and little repetition. For example, the ASAP data set with about 2000 answers per individual question contains no duplicates at all, neither among the correct nor the incorrect answers. These answers usually consist of one or a few full sentences, so it might be not surprising to see that each answer is unique. But even the Powergrading data set with very short answers, –most answers consist of just a few words– where we would therefore expect a higher degree of duplicate answers, still has between 20 and 582 unique answers per prompt (where each prompt has a total of 698 individual answers) and for most prompts there are more unique correct than incorrect answers. This means that it is not feasible for a teacher to straightforwardly and preemptively define all possible wordings for a correct solution to an SAQ. Therefore, a procedure where correct answers can be identified by simple string matching and all other answers are counted as incorrect will perform poorly.

One way out of the scoring dilemma would be to avoid SAQs in automatic scoring scenarios completely. And indeed, many application scenarios for SAQs could also be modeled through

exercise types that are easy to score automatically, such as multiple choice questions, where the student has to select the correct answer out of several alternatives. However, there are good reasons not to eliminate SAQs from the area of automatic scoring. First, the usage of multiple choice questions bears its own problems. Many teachers doubt that it is an appropriate way of testing a student's knowledge (Davies, 2002) since recognizing a correct answer is in general easier than producing it (as, for example, shown by Laufer and Goldstein (2004) for vocabulary learning). Furthermore, the creation of appropriate distractor items is also non-trivial both for humans and computers (Haladyna and Rodriguez). Additionally, SAQs have a long tradition in classroom-based teaching and teachers use them with purpose: While it would definitely be easier for manual scoring by a teacher to correct multiple choice questions rather than free-text answers, SAQs offer the additional advantage that they elicit natural language responses from the students. They force them to express their thoughts coherently and, in the case of second language learners, to also practice writing in a foreign language.

In contrast, when comparing SAQs to longer free-text student writings, such as essays, the complexity of SAQ answers is restricted because of their length: they are by definition shorter than an essay (Burrows et al., 2014). In the extreme case, an answer can consist of just a single word, but typically contains one or two sentences. This is a level of complexity, content scoring methods can handle (Bailey and Meurers, 2008). In contrast, content-based scoring of essays is a much harder task requiring deeper processing.

Automatic short-answer scoring can be applied with different goals in mind: common to all of them is the aim of generating a *label* (such as a numeric score or a binary correctness decision) for an answer without a human annotator directly labeling it. We have seen in the last section that formative and summative assessment are two applications for short-answer questions. They can also both be the goal of automatic scoring.

The automation of summative assessment, which is used for grading purposes, has typically the general goal of reducing a teacher's workload. This is the type of assessment we model in this thesis. Instead of grading all student answers in, for example, a placement test or a homework assignment, only a subset is labeled to train a classifier. If a test is repeatedly administered (in the case of prompt-specific models) or questions of a similar type are added later (for prompt-independent models), a teacher could only label answers to train a classifier for the first application of the test and re-use an existing classifier later. Such a workload reduction almost always comes with some decrease in grading performance compared to manual scoring, therefore finding a trade-off between the two is important in such a scenario.

In formative assessment, the recipient of the feedback is the student and not the teacher. A typical application for automated formative assessment is on-line tutoring systems. In such a scenario, learners receive instant feedback on their answers, generated from automatically

assigned labels.

The envisioned application scenario, especially the decision regarding whether summative or formative feedback is required, influences the label set used for ASAS. While numeric scores are useful for a teacher in a testing scenario, they might not provide informative feedback for a student. A student does not profit much from the information that her answer has been scored with 3.5 out of 5 points, but benefits more from a diagnostic label telling her that the answer is missing some concept or contains unnecessary material not asked for in the question (and ideally also stating which concept is missing or which one is incorrect). While such a diagnostic label is helpful for a student, it is unsuitable if a teacher wants to compile an overall score for a number of answers given by one student in a test, where numeric or binary scores would be more suitable.

For some approaches to automatic short-answer scoring, additional material is necessary for the scoring process. All SAQs come obviously with a prompt, i.e., the question and any other material presented to the student. Additionally, as we have seen in the examples before, a so-called *target answer* is often, but not always, explicitly specified. A target answer is a sample solution, typically written by a teacher, expressing a correct answer to the question. Such a target answer has to be provided in some approaches as reference for comparison to the learner answer (especially for prompt-independent approaches as we will see in Section 2.4). In a classroom setting, teachers of course also have a target answer in mind when using an SAQ, i.e., they know what correct answer they expect. But it is not necessarily explicitly written down. For some ASAS corpora, more than one such answer is given for some questions, others contain a single target answer per question. Some corpora do not provide target answers, but come with scoring guidelines instead. One reason is that there are often several conceptually different correct answers for the prompts in these data sets and answers given to these prompts tend to be quite long. Therefore, it is more convenient for the human annotator to have specifications what features of correct and incorrect answers are, respectively (e.g., that a correct answer must address at least two out of a list of several points), instead of giving examples for all conceptually different variants of a correct answer. Such a data set is therefore mostly used with prompt-specific approaches that do not require a target answer.

2.1.3 Suitability of Short-Answer Question Types for Automatic Scoring

This section reviews an existing typology of reading comprehension questions and evaluates which types are suitable for automatic scoring. Not all kinds of short-answer questions are equally well-suited for automatic evaluation. Primarily those questions that ask for verifiable information, instead of personal opinions, are good candidates, as we will see below. It is natural that science or problem solving questions ask for such verifiable information, but the distinction

regarding whether a question asks for facts inferable from the text instead of personal opinions is especially important for the application scenario of reading or listening comprehension.

Day and Park (2005) propose a six-way classification of reading comprehension exercises into *Literal*, *Reorganization*, *Inference*, *Prediction*, *Evaluation* and *Personal Response*. Out of those, only the first three types can be answered using only information from the reading text used in the reading comprehension task: Literal questions "can be answered directly and explicitly from the text" (Day and Park, 2005), such as questions asking for times, dates, locations, etc., e.g., *"When was Peter born?"* For reorganization questions, students need to combine information from different parts of the reading text. For inference questions, the answer is not literally present, but the learner needs to draw an inference based on their own knowledge. For questions of these three types, objectively correct answers exist, and there is typically only a limited number of conceptually different answers per questions, so that they are suitable for automatic scoring.

The other three comprehension types are more problematic with regard to automatic scoring, as often many conceptually different answers exist for questions of these types. Prediction questions require the reader to use "both their understanding of the passage and their own knowledge of the topic and related matters in a systematic fashion to determine what might happen next or after the story ends." Day and Park (2005) give as an example the question *"Do you think they will stay married? Why or why not?"* Such an answer is more problematic to score automatically as there are in nearly all cases very different correct target answers and the correctness of the answer depends not so much on the content as on whether the student presented a plausible justification for his answer. The same holds for evaluation questions where students have to give "a judgment about some aspect of the text" such as in the question *"How will the information in this article be useful for you?"*. Potentially even more problematic are personal response questions such as *"What do you like or dislike about the article?"* (examples taken from Day and Park (2005)). We expect that answers to such questions cannot be automatically scored using typical ASAS methods that either rely on commonalities between correct answers or between a correct answer and a target answer. Each answer will be personal and different from the others, and correctness depends on other factors than just content, such as argumentation. As far as we know, instances of the last three comprehension types indeed do not occur in any reading comprehension data set for automatic short-answer scoring.

After we have seen what ASAS is, that SAQs are used in a variety of educational context and which types of SAQs are suitable for automatic scoring, we will compare it to other natural language processing tasks.

2.2 NLP Tasks Related to ASAS

This section places the ASAS task in comparison to related natural language processing (NLP) tasks, namely those of recognizing textual entailment, paraphrase detection, and question answering. These tasks have in common with each other and with ASAS that they perform semantic analysis of texts.

2.2.1 ASAS and Textual Entailment

ASAS has often been compared or even equated to the task of *recognizing textual entailment* (RTE). We discuss some notable comparisons in this section. We also address the notion of *partial textual entailment*, which has been introduced to express entailment relations on a more fine-grained level, and present approaches targeting this task.

RTE is a widely known task within computational semantics that has attracted much interest through the RTE workshop series that was first established in 2006 by Dagan et al. (2006). The RTE task in its original formulation by Dagan and Glickman (2004) is a binary classification task that assesses whether a *text t* textually entails a *hypothesis h* or not. An entailment relation is given if people "reading *T* would typically infer that *H* is most likely true as well" (Dagan et al., 2013). The two-way task with the labels *Entailed* and *Not-entailed* has been extended to a three-way task involving the labels *Entailed*, *Contradicted* and *Unknown* (Giampiccolo et al., 2007). Annual RTE shared tasks led to a growing community proposing a large number of approaches (cf. Dagan et al. (2013)), and further more fine-grained label sets have been proposed, such as, for example, the one by Dzikovska et al. (2013), which will be discussed in more detail below.

In ASAS, answers are assigned a label that assesses the semantic correctness of an answer. This assessment is often performed in direct comparison to a target answer which specifies what a good answer should look like. In this sense, RTE, like ASAS, is a task in which text pairs are considered and a label is assigned based on the relation between the two texts. Some approaches involving a target answer have compared the ASAS task explicitly to an entailment task between a learner answer and the target answer. We will address these comparisons in the following.

For the c-rater system (Sukkarieh and Blackmore, 2009), one of the most prominent ASAS systems – see Section 2.4.1 for a more detailed discussion of the approach – the authors explicitly state that they see the c-rater scoring task as a textual entailment problem, where a correct answer is either "a paraphrase (or) an inference up to a context". More specifically, they check whether a concept necessary for a correct answer is an inference or a paraphrase of a learner answer given the question the learner answer addresses, i.e., they check whether all elements required for a correct answer are indeed inferable from the learner answer and thus are entailed by it.

SemEval-2013 Task 7: The Joint Student Response Analysis and 8th Recognizing Textual

Entailment Challenge (Dzikovska et al., 2013) explicitly brought the tasks of ASAS and RTE together in a combined shared task, following a tradition to combine the RTE challenges with a specific application, such as, e.g., text summarization in RTE-6 (Bentivogli et al., 2010). In SemEval-2013 Task 7, the two perspectives of ASAS and RTE on the same data are expressed by using different label sets: They propose two kinds of label sets that both differ from most of those previously introduced for ASAS: a 5-way set which highlights the educational perspective of the task, and a 2- and 3-way label set which emphasizes on the RTE perspective (see Section 3.6 for the data set released with the shared task). Instead of providing scores or binary true/false classifications, their goal for the five-way task is to provide label categories that could be used in the generation of formative feedback to students, for example in a tutoring system. They use the labels *correct*, *partially_correct_incomplete*, *contradictory*, *irrelevant* and *non-domain*. In their 2- and 3-way task, that focuses on the RTE perspective, they assume that the pair consisting of a question and a correct learner answer should entail the target answer while a pair involving an incorrect learner answer does not. Like Sukkarieh and Blackmore (2009) they argue that it is important to consider the question context in the labeling decision as both learner and target answer might or might not be repeating material from the question. (An alternative to this procedure is *question demoting*, where question material is explicitly removed from answers before automatic grading.) In the 2-way task, Dzikovska et al. (2013) follow the original definition of the RTE task and distinguish between entailed answers and not-entailed answers, where they claim that entailed answers always correspond to correct answers (i.e., those marked as correct in the 5-way task) and non-entailed answers to incorrect ones (all others). In the 3-way variant of the task, they treat the group of contradictory answers separately and thus obtain the three categories *correct*, *contradictory* and *incorrect*. This 3-way split corresponds to the three-way RTE task introduced by Giampiccolo et al. (2007).[3]

All approaches that see RTE and ASAS as closely related have the assumption in common that a learner answer has to be at least as specific as the corresponding target answer. A learner may be over-specific and mention material that is not necessary, but should not omit necessary parts. This view, of course, assumes that target answers constitute a minimal correct answer that only includes absolutely necessary, but no optional parts of a correct answer. It also assumes that information that is added as extra does not contradict correct information and is not suitable to make an otherwise correct answer wrong.

We challenge this perspective on ASAS in Section 4.1 where we annotate the CREG corpus with entailment information and compare it to teacher-annotated correctness labels. We find that

[3] For data sets where the labels use numeric instead of binary scores, the scoring task is not directly comparable to RTE anymore (Mohler et al., 2011), but rather related to the task of detecting semantic textual similarity (see also Section 2.2.2).

the correspondence between ASAS and RTE holds only for most, but not all items in the CREG corpus.

Nevertheless, the two tasks are similar, therefore it makes sense to try to use similar approaches in both of them. Similarity-based and alignment-based approaches are used both in RTE (Dagan et al., 2013) and ASAS approaches (see Section 2.4). Some RTE systems work proof-based by checking whether the hypothesis can be derived from the text using predefined transformation rules (e.g., the Bar-Ilan University Textual Entailment Engine BIUTEE platform by Stern and Dagan (2013)). While such methods are not very common in ASAS, Levy et al. (2013), for example, use BIUTEE for the task of recognizing partial entailment on the SemEval-2013 data.

In recent years, the notion of *partial entailment* has emerged as a way to measure entailment more gradually instead of framing it as a binary decision. When measuring entailment as a yes/no decision, the information is lost whether an answer is just slightly off and "almost entailed" (Levy et al., 2013) or completely wrong or even contradictory to a correct answer. We have also seen the need to indicate answers that are partially correct in the 5-way classification task in SemEval-2013 Task 7, where *partially_correct_incomplete* was used as a label. In an educational context, Nielsen et al. (2009) proposed to break down target answers into individual components, called *facets*. Facets are "fine-grained semantic components" specifying small units of conceptual knowledge in the target answer that a student has to address in a correct answer. They consist of two words connected by the relation between them. Facets are manually derived from modified dependency parses of a target answer. For example, the following four facets are extracted from the target answer *"A long string produces a low pitch"*: *"NMod(string, long)"*, *"Agent(produces, string)"*, *"Product(produces, pitch)"*, and *"NMod(pitch, low)"*. This approach was the first to allow an operationalization of partial entailment for ASAS. Such a representation provides options for giving precise feedback to students about which parts in their answer are incorrect or what parts are missing in their answer. This feature makes the approach especially attractive for tutoring systems, which aim at formative feedback.

The concept of facets has been taken up in a pilot task connected to SemEval-2013 Task 7, targeting partial entailment. For this task, one sub-corpus of the complete SemEval-2013 Task 7 data set was annotated with facets as defined by Nielsen et al. (2009), with the exception that no label for the relation was given. Learner answers were labeled with respect to each target answer facet with one of the three labels *expressed*, *contradicted* or *unaddressed*. (*Contradicted* does not appear in the actual data set. It is unclear whether it never applied or was removed later).

Two systems have been proposed for the detection of partial textual entailment based on facets. Nielsen et al. (2009) manually annotate facets for each target answer in a corpus of science questions by deriving them from modified dependency parses of the answer. They label pairs con-

sisting of a student answer and a facet that is required for a correct answer using one out of eight labels. These labels specify whether a facet from the target answer is expressed, can be implicitly assumed, is contradicted, not addressed, etc. They conduct machine learning experiments with various lexical and syntactic features and assign a label for each student answer-facet pair indicating the student's understanding of that facet. Classification accuracy clearly outperforms a majority baseline.

Levy et al. (2013) address the problem of recognizing partial textual entailment on the SemEval-2013 Task 7 pilot task data set. (They were actually the only participant in the pilot task.) They learn a binary classification stating whether a facet is *expressed* by an answer or *unaddressed*. For the rule-based decision mechanism, three components are used: the first one requires an exact match of the two lemmas of the facets in the bag-of-words of the student answer. The second one additionally considers lemmas within a certain WordNet similarity range as matches. The third one uses syntactic inference by feeding both the student answer and the subtree from the target answer containing the facet into BIUTEE (Stern and Dagan, 2012) and accepts as matches those cases where BIUTEE recognizes entailment between the answer and the facet. They find that a combination of all three components works best and that the results for partial entailment can also be used to predict binary correctness labels for the data set, based on the heuristic that an answer is correct if all facets are expressed. However, they find that this heuristic works only to some extent: If they use gold-standard annotations in the data set stating for each facet whether it is expressed in the respective answer, they achieve an F_1-Score between 0.71 and 0.77 depending on the exact data set used. This confirms our findings in Section 4.1 that teachers often score answers missing some information specified in the target answer (in this case via facets) as correct.

A drawback of approaches working with facets is that that they require the manual annotation of facets in the target answer and it is questionable how much effort it is for teachers to formulate facets instead of target answers or how facets can be automatically extracted.

2.2.2 ASAS and Paraphrase Detection/Semantic Textual Similarity

Paraphrase detection and detection of semantic textual similarity are two related tasks that are also closely related to ASAS. In *paraphrase detection*, the task is to decide whether two sentences are paraphrases of each other, i.e., whether or not they convey the same meaning (Dolan et al., 2004). Some approaches to ASAS compare a learner answer to a target answer, and learner answers paraphrasing the target answer are correct answers. For these approaches, the close relation between ASAS and paraphrase detection is obvious. As mentioned above, we have seen that the paraphrase relation is a special case of entailment. A paraphrase relation to the target answer is therefore a sufficient but not a necessary condition for a correct answer. A

correct learner answer may contain additional material beyond the target answer or may be more specific than the target answer.

In the literature, many approaches considered the ASAS task as the task to detect paraphrases of the target answer in the learner answer, such as the c-rater system (Sukkarieh and Blackmore, 2009). Mohler et al. (2011) compare ASAS to both paraphrase detection and RTE and note as one difference that RTE and paraphrase detection typically ask for a yes-no decision, whereas ASAS asks for a grade on a scale (not always, but particularly for the data set they introduce in Mohler and Mihalcea (2009)). For example, the scores in the Mohler & Mihalcea data set vary from 0 for a completely incorrect to 5 for a perfect answer.

Similar to teachers and researchers favoring numeric scores for certain tasks in ASAS over binary decisions, the paraphrasing community also measures the relatedness between two sentences gradually. The task of detecting *semantic textual similarity* (STS) was introduced at SemEval-2012 (Agirre et al., 2012) and was the first to ask for a numeric score instead of a binary yes/no decision. In STS, a system has to rate the semantic similarity of two sentences with a score between 0 (the sentences are on different topics) and 5 (they are completely equivalent). For ASAS data sets with numeric scores, the ASAS task is closely related to that of STS: although a numeric score for ASAS should not explicitly assess the semantic similarity to the target answer but the correctness of the learner answer, it does so implicitly since the target answer is one instance of a perfect answer. If we assume that there is conceptually only one target answer to a question then the correctness of a learner answer expresses the semantic similarity to this target answer. Thus in both STS and in ASAS with numeric scores, two sentences or short texts are compared and their relation is numerically scored, based on how semantically similar they are.

Both STS and consequently STS-based approaches to ASAS are inherently insensitive to any directionality between the two texts, i.e., it is unimportant whether one sentence or answer is more or less specific than the other. Approaches from STS were used for ASAS by Mohler and Mihalcea (2009) and Mohler et al. (2011), where the overall textual similarity is computed based on aggregated word similarities and based on graph-based alignments between student and target answer (see also Section 2.4.2).

Part of the problem with the undirected and sometimes hard-to-interpret relation in STS was remedied by the extension of the STS task to that of *Interpretable STS* (ISTS), where an additional layer is added on top of the similarity score that explains why two sentences are related or unrelated (Agirre et al., 2016). This layer consists of alignments between chunks in both texts annotated with both a similarity score between 0 and 5 and a label indicating whether two chunks are (i) equivalent, (ii) in opposition, (iii) similar or (iv) whether one is more specific than the other. This fine-grained understanding of textual similarity brings the task close to the facet

approach by Nielsen et al. (2009) and the partial entailment pilot task by Dzikovska et al. (2013), both discussed in the previous section.

A different approach to making paraphrase detection more fine-grained is the notion of *partial paraphrases* or *paraphrase fragments*, i.e., paraphrasing relations on a sub-sentential level. Instead of scoring the degree of paraphrasing numerically, the part(s) in a sentence standing in a paraphrase relation with parts of another sentence are identified. This method – similar to ISTS, but in contrast to STS – has the advantage of being interpretable, i.e., it can be leveraged for providing feedback. In Section 5.4, we present an ASAS approach that finds partial paraphrases between learner and target answers in CREG using methods to extract paraphrase fragments from comparable corpora.

2.2.3 ASAS and Question Answering

ASAS also has some commonalities with *question answering* (QA). In QA, a system responds to a user's question, also called *query*, by providing one or several answers. This task has been pursued in several evaluation campaigns such as the TREC QA track (Voorhees and Harman, 2005) since 1999 or the CLEF question answering campaign introduced by Magnini et al. (2003). The specific settings vary across years. For some challenges only one answer was allowed per query, for some up to three or five. Sometimes answers were allowed to be chunks of text with a maximum size, for other tasks answers addressing the question exactly were required.

The QA scenario bears some clear resemblances to the ASAS task: First, both tasks deal with questions to be answered, and the questions used in the TREC QA track are similar to those in ASAS. In the first TREC QA track, for example, systems were provided with questions that are characterized as "fact-based, short-answer questions" (Voorhees, 2001). Second, in QA, good answers are typically extracted from a set of documents, and one step of QA comprises of an information retrieval step that identifies texts or paragraphs likely to contain an answer to the query. This task of assessing whether a potential answer candidate actually answers the questions is very similar to the ASAS task.

However, there is one fundamental difference between the two tasks. QA is all about finding the answer to a question automatically in a corpus, while in ASAS the task is to check the correctness of an answer given by a learner. That means, a QA system answering a question fills the same role as a student answering to a prompt and it is different from a system that automatically evaluates such answers.

One could argue that QA can be framed as selecting a good answer out of a set of potential answers, i.e., of judging the correctness of answer candidates. However, there is still a crucial difference between QA and ASAS: QA cares more about relevance than correctness of an answer. In QA, there is typically no way of knowing whether an answer found in the data and

seemingly addressing the question at hand is indeed correct or not. If a user poses the query *"What is the capital of France?"* and the corpus queried contains a sentence *"The capital of France is Madrid"*, then verifying the correctness of that answer is not done or at least not the focus in QA. In other words, QA, as the task is defined, focuses on filtering out irrelevant texts not addressing the answer at all, instead of filtering texts that answer the question with information that would be plausible and on-topic but happens to be incorrect. In ASAS, in contrast, most answers address the question and the task is to decide whether the new information provided in the answer is correct or not. This decision is often based on a target answer, which is of course not available in QA. In most ASAS corpora, only a small fraction of answers are off-topic (such as for example an answer like *"The population of France is 66 millions"* given to the example question above) or non-answers, such as *"I don't know."* In our annotation study on textual entailment relations on the CREG corpus, we found that only about 21% of all answers were off-topic. We assume that in corpora not situated in a language learning context, where understanding the question might already be a problem, this rate might be even lower. In summary, the task of QA deals with deciding whether a text answers a question, whereas ASAS checks whether a question is answered in the right way.

A second difference between the two tasks is the type of evaluation used. In ASAS, one typically measures how well automatically assigned scores correlate with human-assigned labels, i.e., it is of interest how well each individual answer is scored. In QA, one is often more interested whether at least one correct answer appears reasonably high in the results list. In evaluation scenarios where several answers are allowed per question, the mean reciprocal rank (MRR) is used to determine the average position in the answer list at which the first correct answer appears (Gillard et al., 2006). Measures from information retrieval that are derived from precision and recall values are also widely used, such as mean average precision.

In conclusion, while both tasks seem related at first glance, some crucial differences prevent the direct application of QA techniques or systems to ASAS. However, one evaluation contest actively pursued the link between QA and questions asked in an educational context. The Entrance Exams Shared Task of the CLEF Question Answering Track (Peñas et al., 2014) focuses on the automatic evaluation of multiple choice questions in university entrance exams. Systems are presented with triples consisting of a reading text, a question about the text and several multiple choice answer options from which the system has to select one. This setting differs from the standard ASAS scenario. Instead of assessing the correctness of an answer against some target answer, the one correct answer has to be selected from a distractor set solely based on the question and the connected reading text. Note that it is also an artificial task with no direct educational application. The system mimics a student's behavior when taking a test; it would typically not be used, e.g., to automatically score a test as the correct answer would be known to

a teacher anyway so that automatic scoring is trivial.

Despite these differences between the tasks, similar methods as in ASAS can be applied. We participated in the task (Ostermann et al., 2014) using alignment methods that are typically applied to pairs of learner and target answers (Meurers et al., 2011c; Ott et al., 2012; Ziai et al., 2012) to align the question with each text sentence and select the sentence best fitting the question. Next, we aligned each answer option with this text sentence to select the answer that best fits this sentence. Doing so, we reached a modest but competitive performance of 0.36 for the c@1 score (Peñas and Rodrigo, 2011). The selection of the text sentence that best fits the query is a step similar to QA. The second step of deciding which answer option is addressing this sentence corresponds more to ASAS, where the text sentence and answer option can be compared to target and learner answer.

2.3 Challenges of Learner and Student Language

Foreign language learners formulate answers to short-answer questions with an imperfect knowledge of the language under consideration. They might have both receptive problems in understanding the prompt and productive difficulties when phrasing their answer. Following Selinker (1972), we use the term *learner language* to refer to the resulting language output. Learner language is challenging for both human and automatic processing. In order to process a learner utterance, a teacher needs to know what the student presumably wanted to express by an utterance. We call the corrected version of the utterance the *target hypothesis* (Ellis, 1994). However, it is often unclear or ambiguous what exactly the target hypothesis for a certain utterance might be (Reznicek and Hirschmann, 2013). To illustrate the problem, consider the following example from a reading comprehension task for language learners:

(2.6)
Text: (. . .)Manche halten ihn für 130 Jahre alt, weil Philipp Griebel im thüringischen Gräfenroda schon 1872 einen Terrakotta-Zwerg machte und in seinen Garten stellte. (. . .) Philipp Griebel hatte eine Terrakotta-Manufaktur. (. . .)
(. . .) Some think that garden gnomes are already 130 years old because Philipp Griebel made a terra-cotta gnome already in 1872 in Gräfenroda in Thuringia. (. . .) Phillip Griebel owned a terra-cottay factory. (. . .)
Question: Wer machte den Gartenzwerg berühmt?
Who made garden gnomes famous?
Target Answer: Philipp Griebel machte den Gartenzwerg berühmt.
Phillipp Griebel made garden gnomes famous.
Learner Answer: Eine Terrakotta-Manufaktur machte den Gartenzwerg. (. . .)
A terra-cotta factory made the garden gnome.
TH1: A terra-cotta factory made the garden gnome famous.
TH2: A terra-cotta factory produced the garden gnome.

[CREG]

The learner answer allows at least two different interpretations, which are indicated by *TH1* and *TH2*. Apparently the learner struggled with the construction *"to make something famous"* in the question. It is unclear whether they wanted to express TH1 but forgot the adjective or maybe thought it could be omitted in a form of ellipsis, or whether they actually misunderstood the question as *"Who made garden gnomes?"* and wanted to express TH2 *"A garden factory produced lawn gnomes"*. Even though both interpretations lead to an answer that substantially deviates from the correct answer given in the target answer, the distinction between the two target hypotheses is, for example, relevant for giving a more detailed feedback that would take into consideration whether the answer is actually off-topic (TH2) or not (TH1).

The problem of non-standard language is of course not restricted to non-native writers, but also occurs for native speakers. We expect that severe grammatical problems that impede the understandability of an answer do not occur as frequently as for non-native speakers. Nevertheless, simple typos might change the semantics of an answer drastically if a misspelling leads not to a non-word, but to a different existing word. Consider the following example from a corpus of US immigration test exams. (It is not specified whether answers are given by native speakers of English or not.)

(2.7) **Question:** What is one right or freedom from the First Amendment of the U.S. Constitution?
Target Answers: speech | religion | assembly | press | petition the government
Student Answer: free excess of religion
[Powergrading]

In this example, it is unclear whether *"excess"* is written on purpose or a misspelling of *"access"*, and a teacher or annotator has to decide which version of the target hypothesis to consider for scoring.

For our experiments using automatic processing (Chapters 5 and 6), we decided not to form an explicit target hypothesis or to normalize answers in most of our experiments. Instead, in our prompt-specific scoring approaches (see Chapter 6), we accommodate for the spelling variance using character n-grams as features in addition to lemma n-grams, which are highly sensitive to spelling variations. Consider as an example the word *experiment* and its misspelling *"expermient"*. Even though the words are different on a lemma basis, they share many n-grams on the character level. We address the issue of explicitly normalizing learner language in Chapter 7, where we investigate the effects that normalization has on higher processing steps at the example of part-of-speech tagging.

2.4 Previous Approaches to ASAS

This section presents previous work on ASAS related to our own studies. There has been a large number of approaches to ASAS in recent years; discussing all of them would go beyond the scope of this overview. An extensive survey of ASAS approaches from first systems in the nineties until 2015, reviewing a total of 35 ASAS systems and approaches can be found in Burrows et al. (2014).

An important dimension according to which our studies are classified is whether they perform *prompt-specific* or *prompt-independent scoring*, i.e., whether one model is learned for each new prompt or whether one model is learned across several prompts from the same corpus. In the following, we structure previous work under this aspect. For many approaches the decision of whether prompt-specific or prompt independent models are used follows implicitly from either the structure of the corpora or from the method used. Prompt-specific models can be leveraged only for scenarios where enough answers for an individual prompt are available. In terms of methods, prompt-specific models allow for lexical features covering the occurrence of specific words, lemmas, dependency triples or n-grams in an answer, while prompt-independent models can't encode lexical material directly in their features, but use surface or semantic overlap with corresponding units in the target answer instead. The occurrence of a specific word such as *"Aachen"* might be a good indicator of a correct answer for one specific question such as *"Welche Stadt war auf Platz eins?"* (*"Which city was in first place?"*) about a text talking about the usage of recycling paper in different German cities, but not for any other question from the corpus.

Scoring models in general can be differentiated into three major categories that specify the method used for scoring: (1) approaches that use hand-crafted patterns or manual scoring rules, (2) approaches using supervised machine learning, (3) approaches using clustering techniques. While this categorization is in general orthogonal to the question of prompt-specificity, some combinations are more plausible than others. Manually crafted rules make most sense for prompt-specific scoring. In such approaches a content expert specifies representations for correct answers to specific questions. While it would be theoretically possible to hand-write rules about overlap with target answers it is more intuitive to specify a correct answer according to its lexical content. Therefore approaches of the rule-based type typically belong to the prompt-specific category. Machine learning approaches occur with both prompt-specific and prompt-independent models, only with different types of models as discussed above. Clustering approaches are to the best of our knowledge exclusively used together with prompt-specific scoring. The rationale in clustering-based approaches is that similar answers are put into the same cluster so that a teacher can score items from the same cluster in just one scoring step. Although this has never

been tested, it seems implausible and impractical that teachers score clusters that are formed not based on their content but on the degree of similarity to a target answer. Hybrid models incorporating more than one of these approaches are of course also possible and occur, such as combining clustering with machine learning, as we will see later in Chapter 6.

We present in the following first prompt-specific and then prompt-independent approaches to ASAS.

2.4.1 Prompt-Specific Models

As detailed above, prompt-specific models occur in all three variants of scoring models in the literature - rule-based, supervised machine learning-based, and clustering-based. We present those approaches most relevant to this thesis in the following.

Pattern- and Rule-Based Approaches

As we have seen, pattern- or rule-based scoring models fall naturally in the category of prompt-specific models: In general, pattern- and rule-based models specify a pattern for a correct answers or define concepts that have to be present in a correct answer. These models require a target pattern or a target concept manually annotated by a content expert, for example by a teacher.

The C-rater system (Leacock and Chodorow, 2003; Sukkarieh and Blackmore, 2009), where "C" stands for concept, is probably the most prominent example of a rule-based ASAS system: In the original version of C-rater a *model* of a correct answer is created by a content expert. A model consists of a number of simple sentences each of which represents a concept necessary for a correct answer. Teachers receive help from an interface when creating their models, for example by selecting suitable candidates from a list of automatically generated synonyms of a word. Both the teacher's model and the learner answer are transformed to a so-called canonical representation. They use for example, coreference resolution, spelling correction and synonym replacement, and bring the sentence into a canonical syntactic form by extracting predicate argument structures. The model is then automatically matched against the learner answer in a rule-based way, determining which parts of the answer constitute a paraphrase of a sentence in the model. In later versions of the system (Sukkarieh and Blackmore, 2009), maximum entropy models are used to learn a probability for a match between a concept in the model and a learner answer using features such as whether required arguments match or not. This learning of a matching algorithm is done prompt-independently. Therefore, this later extension of c-rater moves it to the side of prompt-independent scoring: what is needed is a model for each prompt, however, this can be compared to a very elaborate way of defining a target answer. The actual

learning of a model is then done across prompts.

In the Auto-Marking system (Sukkarieh et al., 2003; Sukkarieh and Pulman, 2005; Pulman and Sukkarieh, 2005), so-called *answer keys*, concise representations of acceptable answers, are used to manually create patterns for correct answers for individual prompts. These patterns are constructed on the basis of a set of 200 training instances per question. A pattern consists of both keywords and syntactic annotations and is meant to represent all potential paraphrases of the same answer key. English factoid questions from GCSE exams are used as a data set. In an attempt to facilitate the process of manual pattern creation, a basic tool that supports teachers in writing those patterns is provided.

They additionally propose to use a k-nearest neighbor classifier using token-based tf-idf-weighted lexical features (Sukkarieh et al., 2003) and conduct further machine learning experiments using ILP, decision trees and a naive Bayes classifier (Pulman and Sukkarieh, 2005), but find them outperformed by their patterns. In Sukkarieh et al. (2004), a semi-supervised approach to bootstrap patterns from existing annotated ones is presented, which is able to find patterns which have 89% agreement with manually specified patterns.

Common to both C-rater and Auto-Marking is that patterns have to be created manually in an elaborate way requiring a trained expert (some help in writing those patterns is always provided). Once patterns for a prompt are created, their performance is indeed usually good, however, this approach has the disadvantage that the patterns cannot be applied to new questions, making it impractical in real-life situations where teachers constantly create new questions. We do not use pattern-based approaches in our experiments.

It is interesting to note that even in recent years, hand-crafted patterns are still used highly successfully: The winning system (Tandalla, 2012) from the Kaggle ASAP competition (see also Section 3.3) also uses hand-crafted patterns per questions in a regression model, where the information of whether a pattern fires is integrated via binary features. These features are combined with a number of other features such as lemma uni- to trigrams, but the hand-crafted patterns seem to be responsible for the system's outstanding performance. There have been approaches to automatically detect such relevant patterns. Ramachandran et al. (2015) extract patterns for the ASAP data set both from the scoring guidelines and top-ranking learner answers using word-order graphs. They automatically enrich their patterns using semantically related words as alternatives in the patterns. These patterns, when plugged into the system by (Tandalla, 2012) perform slightly better than the original hand-crafted patterns.

Supervised Machine Learning-Based Approaches

Machine learning approaches learn regularities directly from the data, without the need for a human expert to identify those regularities. Therefore these approaches are more flexible, as

a model for a new prompt can be learned without human effort in creating patterns. However, also supervised machine learning approaches need to learn from *labeled data* and typically more labeled data yields a better model. For a new question to be treated, human annotation effort is needed to label part of the data for the machine learning algorithm to learn from.

Prompt-specific approaches to ASAS have been mainly developed for the Kaggle ASAP competition. Higgins et al. (2014) provide an overview of the top-performing systems. What these systems (Tandalla, 2012; Zbontar, 2012; Conort, 2012; Jesensky, 2012; Peters and Jankiewicz, 2012) have in common is the usage of lexical features, such as word, lemma and/or character n-grams, often extended by dependency triples. Those features provide information about the lexical content of an answer, as well as the syntactic combination of individual words. An additional feature used by some systems is *answer length*, and this feature often works surprisingly well: longer answers tend to be also better answers.[4]

In addition to their review of the scoring competition, Higgins et al. (2014) also provide their own study which extensively assesses the contribution of both lexical and other not syntactically informed features (words, length, LDA topics, number of spelling errors etc.) as well as syntactic features (dependency triples, n-grams, language model features etc.). There they find an improvement by syntactically informed features (a difference of 0.009 in average Correlation between classification results and human scores). While this improvement is quite small and potentially not of practical relevance, they argue that it might help to improve the system's validity, i.e., "the degree to which it will actually measure the skills and knowledge that items are supposed to test" (Higgins et al., 2014) instead of just measuring word choice.

We will see other supervised machine learning approaches that learn models across several prompts in Section 2.4.2.

Clustering-Based Approaches

The core idea in clustering is to put similar items together in one cluster and different items in separate clusters. In the context of ASAS, items that are semantically similar express the same idea and therefore share the same scoring label. Dissimilar items, expressing different concepts, could either have different labels (e.g., one is correct and one is incorrect) or have the same labels. The second case occurs, for example, if both answers are incorrect, but express different misconceptions on the side of the student. Clustering thus ideally leads to a partitioning of answers into groups that each represent either a correct answer (out of possibly several different correct answers to the question) or one way of getting the answer wrong. Each cluster contains items that paraphrase the same concept in different ways. This cluster-intrinsic property of

[4] It is of course questionable whether it is appropriate to use such a feature that can easily be cheated in a real-life scenario.

containing similar items motivates the usage of clustering techniques in short-answer scoring. Teachers see answers in a structured way and can identify common misconceptions among their students and assign feedback to groups of students whose answers are in the same cluster (Basu et al., 2013).

Clustering-based approaches are mainly used in so-called *computer-assisted-scoring* scenarios where, instead of labeling all learner answers manually, clustering provides a workload reduction in one of the following ways: Either a teacher inspects a cluster and assigns a score *holistically*, or only one or a few items per cluster are labeled and this label is then *propagated* to all members of the cluster.

Clustering for ASAS has not been used frequently. Work has been done, in parallel to our first experiments (Horbach et al., 2014a), by the Powergrading Study by Microsoft in two works (Basu et al., 2013; Brooks et al., 2014), described in the following. The Powergrading corpus (see Section 3.5) has been collected within this project.

Basu et al. (2013) describe their clustering approach while Brooks et al. (2014) focus on the applications of the proposed clustering in an evaluation study with teachers. Basu et al. (2013) use k-medoids clustering, a distance-based clustering approach. It first selects a number of initial medoids randomly and then (a) assigns each item to its closest medoid and (b) recomputes the medoid per cluster. These last two steps are repeated until the algorithm converges. For computing the distance between two answers they learn a pairwise similarity metric. All student answers per prompt are clustered into a fixed number of clusters and subclusters.

The cluster evaluation process is centered around the notion of *user actions* and uses holistic scoring of clusters instead of label propagation: a user's macro-action consists of an annotator (re-)labeling all items in one cluster or sub-cluster with a particular label. A micro-action means labeling an individual item. Basu et al. (2013) assume that a teacher can – apparently without effort – select the best next action, i.e., the one that results in the highest number of correctly labeled items. Evaluation is done in two ways: They evaluate (a) how many actions are needed to label all items correctly and (b) how many items are labeled correctly, after a certain number of grading steps. Additionally, they consider both the setting in which all answers are graded "from scratch" and one where clusters are pre-labeled: If the answer most similar to the target answer has a similarity higher than 0.5 the cluster is labeled as correct otherwise as incorrect. They compare their results to an LDA baseline (Blei et al., 2003) in which answers are assigned to their most probable topic. They find that pre-labeling clusters is beneficial and that their approach outperforms the baseline. We discuss issues with this evaluation setup in more detail in Section 6.2 and propose label-propagation as an alternative.

Brooks et al. (2014) evaluate the usability of an interface based on the above clustering in a real-life setting. They compare two settings: one where answers are presented to the teacher in

clusters and one where answers are presented as a flat list. They found that presenting clustered answers leads to faster and equally accurate scoring and that most teachers find working with clustered answers in general helpful to detect trends in student answers.

Zesch et al. (2015) are the first to compare the benefit of clustering (using label propagation as introduced in Horbach et al. (2014a)) both within a supervised machine learning scenario instead of computer-assisted scoring and in direct comparison to supervised machine learning. Comparability between clustering and supervised machine learning is ensured by using the same feature sets, and evaluations are performed on two data sets with the result that clustering in their experiments only helped for prompts with very short answers. We discuss details of this work in Section 6.3. There, we present clustering experiments using label propagation as an alternative to holistic scoring. We propose ways how parts of the observed performance gap between clustering and supervised machine learning can be overcome by combining clustering with feature selection methods from supervised machine learning.

2.4.2 Prompt-Independent Models

Prompt-independent models fall exclusively into the category of (supervised or unsupervised) machine learning models. They can obviously not make use of manually-created patterns for individual answers as their scoring mechanism and, as explained above, clustering answers to different prompts is not feasible either. Instead, prompt-independent models rely on features that establish how close a learner answer is to a target answer on the semantic or surface level. Most notable for this category of models are those systems created by Meurers and colleagues: The earliest of these systems is the Content Assessment Module CAM by Bailey and Meurers (2008) and Meurers et al. (2011a), on the English CREE data. In their system, learner answers and target answers are aligned on various linguistic levels (tokens, chunks and dependency triples) using different kinds of evidence of equivalence on the token level: tokens which are identical on the surface, share the same lemma, are semantically close, or are of the same semantic type can be aligned. Then, chunk and dependency alignments are created based on these token alignments. The system also provides spell-checking and pronoun resolution and uses pre-alignment filters to remove punctuation and lexical material given in the question. A number of features, such as the percentage of aligned tokens, chunks and dependency triples in both the target and learner answer, as well as the nature of the alignments (e.g., the percentage of token-identical tokens among the aligned tokens) are then used as input for a supervised machine learning classifier.

This model for knowledge-based alignment has been adapted for German and used on the CREG corpus (Meurers et al., 2011c; Ott et al., 2012; Ziai et al., 2012). We re-implemented this model and use it as a baseline in some of our experiments on the CREG data, discussed in more detail in Section 5.2. Common to these models is that they build on the similarity between target

and learner answers on various levels and across individual prompts and use them as features in machine learning.

The only deep semantic approach to short-answer scoring known to us has also been developed for CREG and is described in Hahn and Meurers (2012). They use Lexical Resource Semantics (LRS) as a semantic formalism, which is a formalism enabling arbitrary degrees of underspecification, and a syntax-semantic interface using atomic dependency information. In effect, this guarantees that some kind of semantic representation is computed for any (grammatical or ungrammatical) input expression. The LRS representations for target and learner answer are aligned, and, like in the previous approach, alignment features are extracted and used by a classifier. With this approach, they reach state-of-the-art accuracy of 86.3 % on this corpus.

Mohler and Mihalcea (2009) propose a form of unsupervised ASAS that is thus inherently prompt-independent. Their intuition is that the more similar a student answer is to the corresponding target answer, the better the score for that answer. Their approach is thus a direct application of textual semantic similarity as described in Section 2.2.2. They use text-to-text similarity measures to assess the similarity between learner and target answer on a corpus of computer science questions (see Section 3.6 for a description of the data set). They assess the quality of their method by computing the correlation between the normalized individual similarity scores and the gold-standard score of the answer (that is annotated on a scale from 0.0 to 5.0 points). They use both corpus-based and knowledge-based similarity measures. For the corpus-based measures, they leverage eight pairwise word similarities using WordNet. For corpus-based measures they use (a) latent semantic analysis (LSA) (Landauer et al., 1998) trained on either Wikipedia or the British National corpus and (b) explicit semantic analysis (Gabrilovich and Markovitch, 2007) on Wikipedia where each Wikipedia article stands for a concept and a dimension in the resulting vector. Pairwise word-similarities are computed based on cosine similarity between vectors. They derive text-to-text similarity by aggregating the pairwise word similarities: for each content word in the learner answer they determine the similarity of the most similar word in the target answer and return the sum of those individual similarities normalized by answer length as the overall similarity score. They find that several similarity measures work well, while the best performance is reached using LSA trained on a domain specific corpus. They further address an interesting sub-problem when comparing learner to target answers: there might be good learner answers with low similarity to the target answer that would receive an unjustified low score. They tackle the problem by extending the vocabulary of the target answer with words from high-ranking student answers in a bootstrapping-like fashion, and show that they can further improve scoring performance by doing so. This model works surprisingly well, given that no training is required (they report a correlation of 0.50 between their scores and human annotations for the best performing configuration) and that syntactic information is not used at

all. The latter means that the sentence pair *"man bites dog"* and *"dog bites man"* has a perfect similarity of one.

Mohler et al. (2011) address this problem by integrating syntactic information into another approach that uses supervised machine learning instead of just leveraging one similarity score in a completely unsupervised fashion. They produce dependency graphs of both the target answer and the student answer and compute similarity scores between pairs of nodes, with one node from the student and one from the target answer. They use semantic, syntactic and lexical features on the nodes and various types of dependency subgraphs, with a scoring function trained on a small set of hand-aligned pairs. Based on these pairwise node similarities they produce an optimal alignment of nodes in the learner and target answer and extract a number of features from this alignment that takes the strength of the alignment into consideration. These features are combined with a variety of bag-of-words style semantic similarities, similar to the individual scores used in Mohler and Mihalcea (2009). They train a support vector machine on these features and show that they outperform the previous unsupervised approach.

This model is similar to the alignment-based model by Meurers et al. (2011a) and Meurers et al. (2011c) in that it uses features extracted from alignments between learner and target answers. The difference is that the Meurers model uses alignment features mainly based on token- and chunk-level and on individual dependency triples, while Mohler et al. (2011) use full dependency parses as the basis for one important feature block. These differences in the methods fit the respective data sets: the CREG corpus consists of answers given by language learners in a reading comprehension scenario, therefore comparatively little lexical and syntactic variation can be expected as language learners are restricted in their expressiveness by their language level. In contrast, the Mohler and Mihalcea data set consists of answers given by native speaking students to computer science questions where no reading text is available and thus lifting material verbatim from a text can not occur.

Approaches to partial entailment as discussed in Section 2.2.1 are also exclusively prompt-independent, as are the participating systems in the SemEval-2013 Task 7 (Dzikovska et al., 2013).

We propose in Section 5.4 a new option for an alignment-based ASAS approach by learning statistical alignments between learner and target answer by creating parallel corpora consisting of learner answers on the one and target answers on the other side and using alignment methods from statistical machine translation. Additionally, we consider alignments between answers and the reading text on the CREG corpus in Section 5.2, where we check whether learner and target answer refer to the same sentence in the reading text.

2.5 Summary

In this chapter, we have seen that short-answer questions are a commonly used exercise type whose automatic evaluation is both desirable and challenging. It is desirable because short answer questions elicit natural-language answers that cannot be elicited by other, easier to score, exercise types such as multiple-choice answers. It is challenging because of the infinitely many options language has to present the same thought. An additional challenge arises from learner language, i.e., language with deviations from the standard both in terms of spelling and grammar.

We have shown that automatic scoring bears resemblances to other fields in NLP most noticeably to detection of textual entailment and paraphrases. Finally, we reviewed the previous work in ASAS most relevant to our studies. After this overview of the nature and methods of automatic short-answer scoring, we give a review of data sets for the task in the following chapter.

3 Data Sets

After we have established what automatic short-answer scoring is and have given background on both the NLP and educational background involved in ASAS, we will in this chapter have a closer look at data used for the task. We use different corpora in our studies: the Corpus of Reading Comprehension Exercises for German (**CREG**, Ott et al. (2012); Meurers et al. (2011c)), the Automated Student Assessment Prize (**ASAP**) data set from the Kaggle competition[1], Microsoft's Powergrading corpus (**PG**, Basu et al. (2013)) and data collected from placement tests for German-as-a-foreign-language classes at Saarland University (**Laempel**, Horbach et al. (2014a)). We additionally offer a survey of other widely-used ASAS corpora that we do not use in our experiments: the Corpus of Reading Comprehension Exercises for English (CREE, Bailey (2008); Bailey and Meurers (2008); Meurers et al. (2011a)), the Student Response Analysis data set (SRA, (Dzikovska et al., 2013)) and the Mohler and Mihalcea corpus (Mohler and Mihalcea, 2009).

3.1 Properties of ASAS Data Sets

Properties of the corpora, the task and the kind of learners the corpus addresses are important when selecting the appropriate corpus for a study. Table 3.1 presents relevant characteristics of the most widely-used ASAS corpora. The information regarding whether answers have been given by *foreign language learners or native speakers* provides indicators about the type of language we can expect in terms of deviation from the standard grammar and spelling. The *language* the answers are given in is another important factor; it influences the choice of applicable NLP tools. We consider both corpora in English and German. The *tasks* in the corpus (e.g., science questions, reading or listening comprehension, etc.) specify the domain of the prompt. The task also determines whether additional material such as a reading text for reading comprehension scenarios is available or not. The *grading labels* for a data set specify which types of human annotation are present in a data set: such labels can be binary correctness labels, numeric scores or more detailed diagnostic categorical labels, which specify, e.g., whether an answer contains additional content or whether content is missing. The label set is also correlated to the

[1] https://www.kaggle.com/c/asap-sas

Corpus	specific for lang. learners	language	task	grading labels	target answer available?
CREG-1032	yes	German	reading comprehension for language learning	binary & diagnostic	yes
ASAP	no	English	sciences, biology, reading comprehension	numeric $(0 - 2/3)$	no (only scoring guidelines)
Laempel	yes	German	listening comprehension for language learning	numeric $(0.0 - 2.0)$	yes
Powergrading	?	English	immigration exams	binary	yes
CREE	yes	English	reading comprehension for language learning	binary & diagnostic	yes
SRA	no	English	science questions	entailment labels (binary/diagnostic)	yes
Mohler & Mihalcea	no	English	computer science questions	numeric $(0.0 - 5.0)$	yes

Table 3.1: Overview of ASAS corpora, the four corpora in the first block are used in our experiments.

assumed usage scenario for the prompts in the corpus. Binary and numeric scores are often used in summative feedback to the teacher, e.g., for scoring a test. Diagnostic scores, in contrast, are often used when formative feedback is given to students, e.g., in an on-line tutoring system (see Section 2.1.2).

Corpus	answers	prompts	tokens per answer		
			avg.	min.	max.
CREG-1032	1032	177	11.1	5.0	45.8
ASAP	33320	10	48.4	26.5	66.1
Laempel	1668	21	5.4	1.6	10.0
Powergrading	6980	10	3.9	1.5	7.8
CREE	566	62	23.4	5.7	68.1
SRA	5239	182	12.3	3.2	41.9
Mohler & Mihalcea	630	21	20.7	6.6	36.2

Table 3.2: Corpora Statistics. Tokens per answer are counted individually per prompt and the average, minimum and maximum across all prompts is reported.

For choosing applicable algorithms for automatic scoring, it is important to consider how many learner answers are overall available in a corpus, and how many answers are available per individual prompt. The latter is relevant for determining whether prompt-specific methods such

as clustering are possible, since such approaches typically require a larger number of answers for the same prompt compared to prompt-independent methods, where data from different prompts is used as training data. To characterize the individual data sets further, we also report the average length of an answer (measured in tokens) and report minimum, maximum and average across all prompts in a corpus. This information is presented in Table 3.2.

> Note that spelling and grammar error in the examples are shown as they appear in the data (both in the learner answers and sometimes also in target answers and reading texts). English translations of German data intend to reflect errors in the original data by the translation. (Translation done by Alexis Palmer and the author.)

Lexical Diversity of ASAS corpora

Lexical diversity is a measure that determines the variance in wording in a text, and is typically used to measure the richness of vocabulary in a single text, for example in essay scoring (Mellor, 2011). As mentioned earlier, one challenge for ASAS is language variation. Therefore, we investigate here the lexical variance of the answers given to specific prompts of the different data sets we work with. That means, we measure whether the different answers to one prompt are very repetitive or whether they show a variety of different formulations. By averaging over the prompts in specific data sets, we can highlight the differences between the data sets used in this thesis.

Lexical diversity is a measure on texts, not on collections of texts. Therefore we measure lexical diversity on just one (pseudo-)text per prompt, for which we randomly concatenate all learner answers to that prompt together. Lexical diversity is typically expressed by the type-token-ratio (TTR) (Templin, 1957), where the number of individual types is divided by the overall number of tokens in a text. This measure is, however, only meaningful for comparison of texts of similar length (Schmitt, 2010); the longer a text becomes, the less likely it is that it will contain a new word. Our corpora vary considerably with respect to the number of answers per prompt (see Table 3.2), so that TTR in its original form is not suitable for comparing them.

We decided to use a variant of standardized TTR using random sampling instead of continuous text to avoid artifacts from concatenating learner answers randomly. We consider 20 tokens per text to accommodate prompts with very few answers, and sample 10000 times.

TTR measures are sensitive to spelling mistakes: Frequent misspellings lead to a high lexical variance on the token level, as e.g., *"experiment"*, *"expermient"*, and *"experimant"* are all counted as different tokens. There is no way of separating these effects from true lexical diversity, where different lexical items were intended and not produced by mistake, without a complete normalization of all data. As a way to mitigate these effects, we additionally com-

Corpus	TTR			character 3-gram TTR		
	avg.	min.	max.	avg.	min.	max.
CREG-1032	0.795	0.522	0.966	0.952	0.85	1.000
ASAP	**0.877**	0.85	0.893	**0.969**	0.963	0.973
Laempel	0.716	0.365	0.892	0.874	0.605	0.969
PG	**0.573**	0.255	0.874	**0.832**	0.564	0.971
CREE	0.781	0.516	0.898	0.945	0.867	0.983
SRA	0.771	0.56	0.869	0.934	0.854	0.965
Mohler & Mihalcea	0.831	0.693	0.9	0.956	0.897	0.977

Table 3.3: Lexical Diversity of ASAS corpora. We compute the diversity on token and character trigram level per prompt in each corpus and report minimum, maximum and average of these values. Highlighted in bold are the minimum and maximum average complexity across all corpora.

pute also a variant of TTR not on token level, but on the level of character n-grams. (n-grams are only computed within individual answers and not across answer boundaries within a pseudo-text). While *"experiment"* and *"expermient"* are two different tokens, they share many character n-grams. We therefore consider character n-grams as a measure more robust to spelling mistakes and report TTR values of character 3-grams (again using a sample size of 20 n-grams, selected randomly 10000 times per text).

Results for both TTR on token and on character trigram level are presented in Table 3.3. We compute the TTR value for each prompt in each data set separately and report for each corpus both the minimum and maximum prompt as well as the average across all prompts.

We can see a wide variety of TTR scores in the data: ASAP is on the most diverse side showing very high TTR scores, while PG is the least diverse, both in terms of token and character n-gram-based TTR. We can also see that the order of the seven corpora, whether measured under averaged lexical or character 3-gram TTR is stable. For some corpora, we can see that the span between minimal and maximal TTR values is higher than for others, especially the Laempel and Powergrading data shows such a high variance. This can be explained by the design of the corpus that contains for both corpora prompts that are basically asking for a one-word answer with little variation in the data as well as prompts asking for longer answers with higher variability.

In the next sections, we present each corpus separately in detail: The corpora are closely associated in the literature with specific ASAS techniques. In terms of methods, alignment-based scoring is strongly associated with CREG and CREE, clustering with the PG corpus, and

the usage of sentence similarity measures and graph-based alignment models with the Mohler & Mihalcea data set. ASAP and SRA are used in scoring competitions. We discuss these approaches in Section 2.4.

3.2 CREG

CREG, the Corpus of Reading Comprehension Exercises in German, is the main resource for short-answer scoring on German learner data. It has been collected by the University of Tübingen as described by Meurers et al. (2011c) and Ott et al. (2012). The corpus consists of a variety of reading texts, reading comprehension questions about the texts, and target answers as well as a number of learner answers for each question. Learner answers were given by learners of German as a foreign language in the US. Reading texts cover a variety of genres such as newspaper-style articles, letters, advertisements, short stories, dialogs, interviews, etc. The texts are partially simplified or taken from textbooks, and a few texts come with vocabulary annotations in English or even have grammar exercises inserted (such as gaps where the appropriate verb form has to be filled in). Figure 3.4 gives an example of a reading text, one of the connected questions with its target answers and some exemplary learner answers.

The corpus was collected in collaboration with two universities in the US: University of Kansas (KU) and Ohio State University (OSU). The corpus contains data from learners at all levels. Most learner answers were hand-written by students on paper, while some were directly typed. Hand-written answers were transcribed, and all answers were scored by two independent annotators with teaching experience in German (not always the same two annotators for all answers from one university) based on the meaning, i.e., the semantics of the answer, while ignoring spelling or grammatical errors.[2] The annotators assigned two types of labels: a binary label indicating the correctness of an answer and a more fine-grained diagnosis label. This diagnosis label can assume any of the following five values: *correct*, *missing concept*, *extra concept*, *blend*, and *non-answer*. We do not go into detail about this diagnosis label here; inter-annotator-agreement for the diagnosis label is poor and we thus use only the correctness label in our experiments. Various additional meta-data have been collected for this corpus which we also do not use and therefore do not discuss here.

The complete corpus CREG-17k (Ott et al., 2012) consists of about 17000 learner answers. In our experiments for prompt-independent scoring, we use the balanced CREG-1032 subcorpus (Meurers et al., 2011c) containing 1032 learner answers in order to ensure comparability with

[2]Note that also some of the target answers and even a few of the reading texts contain spelling errors. This happens, however, to a much lesser extent than for learner answers and we consider all material that is not a learner answer to be well-formed. In Chapter 7, we show that the POS tagging performance on the reading texts in fact matches that of newspaper data.

TEXT: SCHLOSS PILLNITZ
Das Schloss, das im Osten Dresdens liegt, ist für mich das schönste Schloss in Dresdens Umgebung. (...) Eine besondere Attraktion im Park ist die Kamelie . Die mittlerweile über 230 Jahre alte und 8,90 m hohe Kamelie bekam 1992 ein fahrbares Haus, in dem Temperatur, Belüftung, Luftfeuchte und Beschattung durch einen Klimacomputer geregelt werden. In der warmen Jahreszeit wird das Haus neben die Kamelie gerollt. Während der Blütezeit von Mitte Februar bis April trägt sie zehntausende karminrote Blüten. Ableger der Pillnitzer Kamelie werden jedes Jahr in begrenzter Zahl während der Blütezeit verkauft, dann ist ein Besuch besonders lohnend.

This palace, which lies in the east of Dresden, is to me the most beautiful palace in the Dresden area. (...) One special attraction in the park is the camellia tree. In 1992, the camellia, which is more than 230 years old and 8.90 meters tall, got a new, movable home, in which temperature, ventilation, humidity, and shade are controlled by a climate regulation computer. In the warm seasons, the house is rolled away from the tree. During the Blossom Time, from the middle of February until April, the camellia has tens of thousands of crimson red blossoms. Every year, a limited number of shoots from the Pillnitz camellia are sold during the Blossom Time, making it an especially worthwhile time to visit.

QUESTION:
Ein Freund von dir möchte sich die alte Kamelienpflanze ansehen. Wann sollte er nach Pillnitz gehen und warum gerade in dieser Zeit?
A friend of yours would like to see the historic camellia tree. When should he go to Pillnitz, and why exactly at this time?

TARGET ANSWERS:

- Von Mitte Februar bis April ist die Blütezeit.
  From the middle of February until April is the Blossom Time.
- Im Frühling trägt die Kamelienpflanze zehntausende karminrote Blüten.
  In spring the camellia has tens of thousands of crimson red blossoms.

LEARNER ANSWERS:

- [correct] Er sollte Mitte Februar bis April gehen, weil die alte Kamelienpflanze zehntausende karminrote Blüten trägt.
  He should go from the middle of February until April, because then the historic camellia has tens of thousands of crimson red blossoms.
- [incorrect] Der Pillnitzer Kamelie werden jedes Jahr in begrenzter Zahl während der Blütezeit verkauft.
  Every year, a limited number of Pillnitz camellia are sold during the Blossom Time.
- [incorrect] Alles Jahr wegen dem Temperatur und Luftfeuchte durch einen Klimacomputer geregelt werden.
  All year round because temperature and humidity are controlled by a climate regulation computer.

Figure 3.4: Example from CREG consisting of a reading text with question and answers

previous work on this data set (cf. Section 2.4.2). The corpus contains 177 individual questions (some of which are identical so that there is a total of 167 unique questions), 327 target answers (272 unique answers) and 1032 learner answers (all unique, if we consider the first transcript, which we are using for most of our experiments). Half of the answers are graded as correct and half as incorrect. The questions target 30 texts (12 for OSU, 18 for KU, for 11 KU questions no text id is specified). When the corpus contains more than one target answer for a question, it also provides annotations that link most learner answer transcripts to exactly one best-fitting target answer.

3.3 ASAP

The data set for the Automated Student Assessment Prize (ASAP) competition[3] contains 10 individual prompts of very different kinds; they include reading comprehension, science and biology questions. All data is in English. Figure 3.5 shows examples of questions and answers in ASAP. Note that the complete prompts tend to be very long (up to two pages of reading text) and are not displayed here for space reasons.

This data set stands out from the others for three reasons: one is the length of individual answers, on average 50 tokens, which makes some of the answers look more like short essays than answers to short-answer questions. The second unique feature is the very high number of individual answers per prompt. Third, the data set comes with scoring guidelines instead of target answers. Most prompts simply specify a scoring rubric for each number of points. Consider as an example the following scoring guideline that specifies requirements for a learner answer with two points (out of a maximum of three):

> *The response is a proficient answer to the question. It is generally correct, complete, and appropriate, although minor inaccuracies may appear. There may be limited evidence of elaboration, extension, higher-order thinking, and relevant prior knowledge, or there may be significant evidence of these traits but other flaws (e.g., inaccuracies, omissions, inappropriateness) may be more than minor.*

Such a scoring rubric is in its nature substantially different from a target answer and can, of course, not be used to directly compare the student answer to. For the two biology prompts, the scoring guidelines list key elements of a good answer and then state how many such points a student needs to address for a certain score (e.g., *"Key Elements: -mRNA exits nucleus via nuclear pore. -mRNA travels through the cytoplasm to the ribosome or enters the rough endoplasmic reticulum. (...)"*)

[3] https://www.kaggle.com/c/asap-sas

QUESTION 1: After reading the groups procedure, describe what additional information you would need in order to replicate the experiment. Make sure to include at least three pieces of information.

STUDENT ANSWERS:

- **Answer (3 points):** Some additional information you will need are the material. You also need to know the size of the contaneir to measure how the acid rain effected it. You need to know how much vineager is used for each sample. Another thing that would help is to know how big the sample stones are by measureing the best possible way.
- **Answer (1 point):** After reading the expirement, I realized that the additional information you need to replicate the expireiment is one, the amant of vinegar you poured in each container, two, label the containers before you start yar expirement and three, write a conclusion to make sure yar results are accurate.
- **Answer (0 points):** The student should list what rock is better and what rock is the worse in the procedure.

QUESTION 8: During the story, the reader gets background information about Mr. Leonard. Explain the effect that background information has on Paul. Support your response with details from the story.

STUDENT ANSWERS:

- **Answer (2 points):** Paul sees himself in Mr. Leonard. They both can't read but both are good at track.
- **Answer (1 point):** When Paul finds out about Mr. Leonard's background, Paul can relate well to him. Paul realize's that Mr. Leonard had given up his own time to help Paul with track and now Paul says it's his turn to help Mr. Leonard.
- **Answer (0 points):** He (the narrator) finds out that Mr. Leonard was once a star athelete at his college but dropped out because of grades.

Figure 3.5: Example questions and answers from ASAP.

The data comes in the form of a training corpus (17208 answers) and an evaluation corpus (5732 answers, the "public leaderboard data set"), leading to an average of 2340 answers for each of the 10 prompts. This high number of items per prompt makes the corpus especially suitable for prompt-specific scoring (cf. Section 2.4.1).

The corpus has two more noteworthy features: First, label distributions are very skewed for some prompts. Second, inter-annotator agreement between the two expert annotators is relatively low for some prompts. Also, it is often not intuitively easy to understand why an answer has a certain score. Table 3.6 shows the label distributions and the inter-annotator agreement on the training data using linearly weighted kappa for each of the ten prompts.

		label distribution				
prompt	**#answers**	**0.0**	**1.0**	**2.0**	**3.0**	**kappa**
1	2229	483	597	699	450	0.94
2	1704	236	426	612	430	0.91
3	2214	530	1235	449	-	0.76
4	1952	761	1035	156	-	0.75
5	2393	1853	429	64	47	0.95
6	2396	2013	216	101	66	0.96
7	2398	1227	600	571	-	0.97
8	2398	725	622	1051	-	0.86
9	2397	585	973	839	-	0.83
10	2186	371	1036	779	-	0.88

Table 3.6: Label distribution and inter-annotator agreement (quadratically weighted kappa) between both annotators for the ASAP data set

3.4 Laempel

The Laempel data set consists of answers to listening comprehension exercises. The listening comprehension task is one component of a placement test for German-as-a-foreign-language courses at Saarland University (the other two parts being a grammar test and a c-test, both in the form of gap-filling exercises). Students listen to a prerecorded audio segment and answer questions about the text they have heard.

The data collection has been done online via a web-based language-learning platform. The questions asked are of various types, ranging from those looking for a single word answer (e.g., Higgins et al. (2014) Where is she from?) to questions asking for longer answers, such as questions requesting explanations (e.g., Higgins et al. (2014) Why does she have to leave?). Students are not required to answer questions with complete sentences. Figure 3.7 gives examples of prompts and answers occurring in the data set. Note the variety in form, even among correct answers.

We use individual answers to 21 different questions about 3 audio texts, collected from 98 students. These are data from a placement test administered in August 2013. Together with the textual content produced by the learners, the data consists of target answers provided by teachers for each question and of teacher-assigned grades for each learner answer. Grades are given as numeric values, usually 0.0 for incorrect answers, 0.5 for partially correct answers, and 1.0 for correct answers. Some questions are assigned up to 2.0 points.

This corpus is the only one we use that contains data from listening comprehension exercises. Listening comprehension data differs from reading comprehension data in that it contains a higher degree of orthographic variability, as there are no options to lift material from the text verbatim by sight. Another difference between the listening comprehension data in the Laempel corpus and both CREG and ASAP is length. Where these corpora contain answers that are between 1 and 3 sentences in length, the average length of learner answers in Laempel is typically just one sentence with around 5 tokens, similar to the Powergrading corpus discussed next.

3.5 Powergrading

The Microsoft Powergrading (PG) data set as described by Basu et al. (2013) consists of 20 questions selected from the United States Citizenship Exam, i.e., those questions are normally asked in naturalization interviews for people intending to become US citizens. Example 3.8 shows 2 out of those 20 questions with some correct and incorrect answers.

For the PG data set, written answers were collected using Amazon Mechanical Turk. Although the original nature of an immigration exam suggests that the test takers are mostly non-native speakers, it is not stated by the authors whether turkers are selected according to their native language. The frequency of spelling mistakes suggests, however, that at least some of the turkers were not native speakers. The authors collected a total of 798 responses per question, split into 100 answers for training and 698 for testing. This might seem an unusual split with a large proportion of test data. However, it fits their unsupervised clustering scenario where training data is only used to calibrate a similarity measure and not within the actual machine learning process. See also Section 2.4.1 for an overview of their clustering approach. They selected 10 out of those 20 questions for grading such that they cover a variety of different answer lengths. The answers were manually scored by 3 graders as either correct or incorrect. Each prompt comes with one or several target answers (referred to as *answer keys*) that were also available to the annotators. Additionally, answers were divided by one of the authors into groups of semantically equivalent answers to form a gold standard for the clustering methods they apply. In contrast to CREG and ASAP, but similar to the Laempel data, answers are very short (on average 5.4 tokens per answer); we can observe that both target answers and learner answers are often not complete sentences but rather short phrases. As the prompts consist of just a single-sentence question without any additional reading material, we observe a high spelling variability, because turkers were not able to lift lexical material beyond that occurring in the question from the prompt into their answer. We use in our experiments the part of the corpus that is released with correctness labels: 10 prompts, each one containing 698 answers from the test section.

QUESTION #1:
Als was arbeitet Julian?
What is Julians occupation?

TARGET ANSWER:
Er arbeitet als Mechaniker.
He is working as a mechanic.

LEARNER ANSWERS:
[1 point / correct]
sie arbeitet als mechanicker
She is working as a mechanik
julian ist mechaniker
julian is a mechanic
mechanika
mechanike
[0.5 points / partially correct]
mekanical
mechanical
[0 points / incorrect]
julian arbeitet in bortschaft
julian is working for embrasy
als sekretarin
as a secretary

QUESTION #2:
Was ist für Nitsa am wichtigsten?
What is most important to Nitsa?

TARGET ANSWER:
Für Nitsa ist es am wichtigsten, sich mit ihren Freunden zu treffen.
It is most important to her to meet her friends.

LEARNER ANSWERS:
[1 point / correct]
freunde treffen
to meet friends
sie mag ihre freunde treffen
she likes meeting her friends
sie akzeptiert dass fernseher ein wichtiges medium aber sie bevorzugt am abends mit ihren freunden zu treffen
she accepts that TV is an important medium but prefers to meet her friends in the evening
[0.5 points / partially correct]
mit freundin
with girlfriend
[0 points / incorrect]
sehr wichtich
very importat
mit freunden ins kino gehen
going with friends to the cinema

Figure 3.7: Sample of answers from Laempel given to two individual questions.

QUESTION:
What is the economic system in the United States?

ANSWER KEY:
capitalist economy
market economy

ANSWERS:

- [correct] a capitalist economy
- [correct] a market system
- [correct] caitalism
- [incorrect] a bad one
- [incorrect] democracy
- [incorrect] a combination of capitalism + command

QUESTION:
What did the Declaration of Independence do?

ANSWER KEY:
announced our independence
announced our independence from Great Britian
declared our independence
declared our independence from Great Britain
said that the United States is free
said that the United States is free from Great Britain

ANSWERS:

- [correct] a document declaring our freedom from england.
- [correct] announce the us's independence from britain.
- [correct] announced america's seperation from great britain and explained why.
- [incorrect] 13 of the american colonies were free from british rule.
- [incorrect] allow the united states to become a country
- [incorrect] allowed americans more rights.

Figure 3.8: Example of prompt, answer key and some sample answers for the Powergrading corpus

3.6 Other ASAS Data Sets

There are a number of other publicly available data sets which are also commonly used for ASAS. Although we do not use these in our experiments for reasons detailed below, we describe them here so that this chapter may serve as a survey of commonly-used ASAS data sets.

The Corpus of Reading Comprehension Exercises in English (CREE) (Bailey, 2008; Bailey and Meurers, 2008; Meurers et al., 2011a) contains, similarly to the CREG corpus for German, answers to reading comprehension exercises given by English-as-a-second-language learners. The research on this corpus paved the ground for the alignment-based scoring model by Ott et al. (2012) and Meurers et al. (2011c). Like CREG, CREE also comes with both binary correctness and more fine-grained diagnosis labels. With a total of 566 learner answers for 75 prompts, the data set is too small for our experiments.

The Student Response Analysis Corpus (SRA) used in the SemEval2013 Task-7 (Dzikovska et al., 2013) (see also Section 2.2.1) contains English student answers to explanation and definition questions. The data consists of two subsets: BEETLE II contains data from tutorial dialogs in physics targeting undergraduate students (the original data set contains also factoid questions with very short answers which were omitted for SRA); SCIENTSBANK contains physics questions for US students from grades 3 to 6. Each question (182 in total in the training data) comes with a target answer. The partial entailment pilot task also contains a number of *facets* necessary for a correct answer for the SCIENTSBANK data. The test data additionally contains also unseen questions.

An interesting feature of this corpus is that it is annotated with two related label sets: Student answers are labeled with one of the following five labels that are created in a way to be of use in a tutoring system: *Correct*, *Partially_correct_incomplete*, *Contradictory*, *Irrelevant*, and *Non-domain*. Binary labels were by (Dzikovska et al., 2013) inferred from this label set by scoring only *Correct* answers as correct and all others as incorrect. We have discussed these label sets and especially the proposed mapping of the diagnosis labels to binary labels in more detail in Section 2.2.1 when looking at the relation between short-answer scoring and textual entailment. In Section 4.1, we will test the validity of this mapping on the CREG corpus, where similar label sets have been annotated independently from each other.

We did not use this data set in our ASAS experiments because of the labeling scheme that is targeted more towards tutorial dialogue and because the number of student answers per individual question is relatively small for prompt-dependent scoring (between 34 and 129 answers per prompt in the training data).

The automatic short-answer grading data set by Mohler and Mihalcea (2009) is a data set in English and contains a total of 21 individual prompts as part of an introductory computer science

class at the University of North Texas. Each prompt in the data set consists of a question such as *"What is the role of a prototype program in problem solving?"* Each prompt comes with a target answer and several learner answers, which sum up to a total of 630 learner answers. Two annotators graded the answers for correctness. A numeric score ranging between 0 and 5 points in steps of 0.5 was given to each answer in the corpus as the average of the two annotator scores. Like SRA, this corpus does not address a language learning scenario and the authors do not state whether all learners are native speakers or not, but it seems plausible that many of them are. We do not use this data set in our experiments because of its comparatively small size, both regarding the total amount of answers and the number of answers per prompt.

3.7 Summary

We presented in this chapter the corpora used in our studies. From the range of corpora available, we selected those that fit the respective goals of our studies best. As one main goal of the thesis is to investigate ASAS methods that reduce a teacher's scoring workload (instead of giving feedback to students), we are especially interested in data sets with labels that are meaningful for teachers in testing scenarios. That requires data sets with binary or numeric scores for summative feedback, instead of data sets with diagnostic labels for formative feedback, ruling out the SRA data set.

For investigating relations between learner and target answers as well as between answers and reading texts in Chapter 4, we use CREG, which comes with reading texts and target answers. Our modeling experiments for prompt-independent scoring in Chapter 5 also rely on the availability of target answers and partially also on that of reading texts. Our approaches to prompt-specific scoring in Chapter 6, in contrast, need corpora with large amounts of answers per individual prompt. Therefore, ASAP, PG, and Laempel are used in these studies.

4 Corpus Studies – Exploring the Semantic Relations between Learner Answers, Target Answers and Prompts

In this chapter, we present three annotation studies in order to shed light on the relations between learner answers and the additional pieces of text they stand in relation with. Answers always address a particular question, they can be compared to a potential target answer, and – for reading comprehension exercises – their content mostly originates from a reading text (see Figure 4.1).

When creating a new short-answer question, a teacher has a certain correct solution – or at least criteria for a good solution – in mind. Such an explicitly formulated target answer – if it is available – is used as the basis for automatic scoring in ASAS approaches using similarity or alignment measures between learner answers and target answers (see Section 2.4). Therefore, the relation between the learner answer and the target answer obviously is important; ideally, a correct learner answer is equivalent to the target answer. However, often a certain degree of semantic similarity between target answer and learner answer is already sufficient for a teacher to score the answer as correct. Previous approaches have compared the task of ASAS to that of RTE (see Section 2.2.1) and claimed that it is not enough for a correct answer to be somewhat similar to a learner answer, but that a correct learner answer always entails the target answer or is even a paraphrase of it. We test this view in **Annotation Study 1** by explicitly comparing existing ASAS labels for learner answers and newly annotated RTE labels on answer pairs, each consisting of a learner answer answer and its corresponding target answer.

Answers to short-answer questions are not texts standing in isolation or just in comparison to a target answer - they always stand in relation to the prompt they are addressing. This prompt always contains a question. For some types of short-answer questions, the prompt consists of only the question, such as the computer science questions in the Mihalcea data set (Section 3.6). In the case of reading comprehension, which we consider in the studies in this chapter, the prompt additionally contains the reading text the question is about. The relation between an answer and the reading text is the topic of **Annotation Studies 2 and 3**: We hypothesize that a correct answer to a reading comprehension question is always supported by the reading text, i.e., it is supported by some sentence(s) in the text. From an RTE perspective, a correct answer

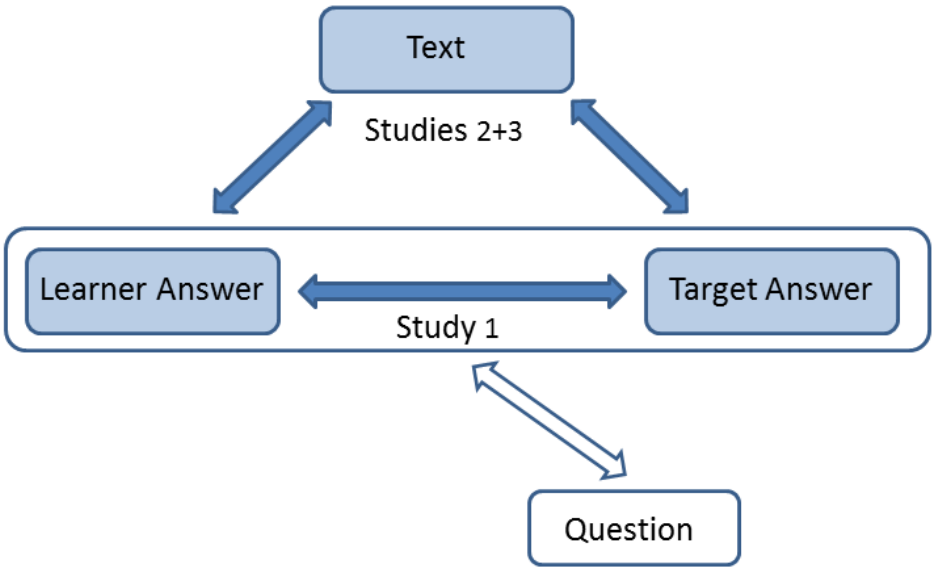

Figure 4.1: Visualization of the relations targeted in the three annotation studies: Study 1 targets entailment relations between learner and target answers. Studies 2 and 3 focus on the relation between answers and the text. Study 2 identifies source sentences for each answer, and Study 3 investigates the entailment relation between those source sentences and the answer.

should be entailed by this text passage. If an answer is not supported by the text, we assume that it is incorrect. At the same time we hypothesize that information in an incorrect answer also occurs somewhere in the reading text. In **Annotation Study 2**, we mark the sentences a learner most likely looked at when creating their answer; we call them source sentences. We do not ask for sentences entailing the answer. Instead, we expect that source sentences are sometimes indeed non-entailing and that some answers are just semantically similar to the source sentence and contain lexical material from it. As the next step, we additionally study the entailment relations between source sentences and learner answers in **Annotation Study 3**.

The corpus of choice for all these annotation studies is the CREG corpus (Ott et al., 2012) because it is, to our knowledge, the only short-answer scoring corpus that contains only reading comprehension questions and for which the reading texts are available.

4.1 Annotation Study 1: Textual Entailment Relations between Learner and Target Answers

It has been often noted that the ASAS task is related – if not equivalent – to the task of recognizing textual entailment (RTE, e.g., Mohler et al. (2011), Sukkarieh and Blackmore (2009), Dzikovska et al. (2013)). For a more detailed discussion see Section 2.2.1. RTE is the task of deciding whether a *text* T textually entails a *hypothesis* H, i.e., whether people "reading T would typically infer that H is most likely true as well" (Dagan et al., 2013). This notion is different from logical entailment, where T logically entails H iff whenever T is true, H is also true. Whenever we talk about entailment relations in this chapter, we mean textual entailment, not logical entailment.

Consider the following example from the CREG corpus, where T corresponds to the learner answer and H to the target answer:

(4.1) **Question:** Warum kam Julchen in die Küche?
Why did Julchen come into the kitchen?
Target answer: Julchen kam in die Küche, weil ihre Eltern einen Lärm machten.
Julchen came into the kitchen because of the noise her parents made.
Learner answer (correct): Julchen kam in der Küche, weil Herr und Frau Muschler vor Lachen außer Atem kamen.
Julchen came into the kitchen because Mr. and Mrs. Muschler got out of breath from laughing.

In this example, the learner answer textually entails the target answer. In a strictly logical sense of entailment, laughing until you are out of breath does not always mean making noise, so that the learner answer does not logically entail the target answer. However, it seems plausible to

most people that laughing in that way makes a lot of noise. Such a learner answer, which is more specific than the target answer and thus entails the target answer, is likely to be scored as correct by a teacher.

While the view that a learner answer is correct if it entails the target answer, and incorrect otherwise (Dzikovska et al., 2013), seems plausible, it has never been tested empirically, i.e., it has never been compared to a teacher's understanding of the correctness of an answer. We believe that teachers do not think in categories of entailment and apply additional concepts of correctness while grading. More specifically, we believe that teachers are somewhat forgiving and score some answers as correct that are less specific than the target answer, but "close enough" to a correct answer. Their understanding of the correctness of an answer is crucial for ASAS as they are the experts creating the gold-standard labels for ASAS data sets. Comparing the correctness labels that teachers assign when grading to RTE labels assigned by linguists on a larger data set will give us insights into the criteria used by teachers to assign scores to an answer and thus on the relation between RTE and ASAS. To this end, we present an annotation study that compares correctness labels from the teachers' perspective on the CREG corpus to RTE labels chosen from a linguistic perspective.

ASAS for reading comprehension in a language learning scenario differs from a standard RTE scenario in two important aspects. First, standard RTE compares two texts in isolation. In contrast, the additional context of the question has to be taken into consideration in order to interpret an answer in the ASAS scenario, e.g., to resolve pronouns and to resolve elliptical answers. Second, the learner answers in CREG are given by non-native speakers. We encounter ungrammatical sentences and orthographic variance that are challenging both for many NLP tools (see also Chapter 7) and for teachers or annotators: It is sometimes difficult to understand what the learner wanted to express with an answer (the so-called *target hypothesis*) and thus to determine the correct RTE label. We address both aspects in our annotation guidelines.

Goal of the Study

The goal of this study is to explicitly assess the relation between the two tasks of RTE and ASAS and test claims that RTE and ASAS are equivalent. To do so, we compare RTE labels, which have been annotated without the correctness or quality of the learner answer in mind, to correctness scores assigned by teachers on the CREG corpus, thus addressing research question 1.2:

RQ 1.2: How closely are the two tasks of RTE and ASAS related?

Understanding this relation better might help to leverage techniques from RTE for the task of ASAS and to gain insights into the way teachers score short-answer questions. In concordance

with previous work that considered the two tasks equivalent, we assume that the two tasks are related. But in contrast to Dzikovska et al. (2013), we expect that the relation is not a direct mapping. Like them, we expect that a learner answer will probably be scored as correct by a teacher if it is a paraphrase of the corresponding target answer, i.e., if it entails the target answer and vice versa. When we move away from such clear cases, there is often only partial conceptual overlap between a learner answer and a target answer that does not constitute entailment. In contrast to the argumentation by Dzikovska et al. (2013), we assume that some of these answers are scored as correct and some are not. (According to them, all these answers should be scored as incorrect.) We assume that the score depends, e.g., on how crucial the missing information in the learner answer is or whether the material that the learner added contradicts the reading text. We expect that especially in the case of partial conceptual overlap, there is no direct one-to-one mapping between RTE and ASAS labels.

Specifically, we want to answer the following questions:

- Is a correct learner answer always - as claimed by Dzikovska et al. (2013) – or at least most of the time a paraphrase of the target answer or does it contain the information from the target answer plus some additional information, i.e., does a correct answer entail the target answer?
- Does an incorrect answer never entail the target answer?
- For those cases where it indeed turns out that a correct learner answer does not entail the target answer or an incorrect answer entails the learner answer, how can we explain them?

Contributions

This study makes the following contributions:

- We provide a fine-grained annotation of the CREG corpus (Meurers et al., 2011c) with seven textual entailment labels that specify the entailment relations between learner answers and target answers.
- We provide an evaluation of our annotations that compares how our label distribution corresponds to the distribution of binary teacher-assigned correctness scores in CREG.

4.1.1 Data and Annotations

We use the CREG-1032 corpus (see Section 3.2). This choice was based on the following considerations: First, we need a corpus containing explicit formulations of target answers in order

to look at the RTE relation between learner answers and target answers. CREG provides them. Second, we are also interested in the entailment relations between reading texts and answer (and investigate this relation in Section 4.3). Therefore, we need a reading comprehension corpus with reading texts available. CREG is the only corpus combining all those features, so we use it for this and the following studies. Also, CREG comes with a fine-grained diagnostic label set in addition to the binary correctness labels that bears similarity to the RTE labels themselves. It contains the five labels *correct*, *missing concept*, *extra concept*, *blend*, and *non-answer*. Especially the labels *extra concept* and *missing concept* suggest a correspondence to an entailment relation that is not a paraphrase relation between learner answer and target answer and the other way round. We also compare to this five-way label set.

We extract pairs of learner and target answers from the CREG-1032 subcorpus. Whenever there is more than one target answer available in the corpus for a specific question, we use the one annotated as the best-fitting target answer in the corpus.

Existing RTE Label Sets and our Annotation Scheme

Our label set is based on and inspired by several existing label sets from RTE and ASAS. We have combined and adapted them to meet the requirements of our annotation scenario. For an overview of the correspondences between existing label sets and ours see Table 4.2. In the following, we first review the existing label sets briefly and then present our own labels.

The original formulation of the RTE task contained just two labels: *entailment* and *non-entailment* (Dagan and Glickman, 2004); later, it has been extended by splitting non-entailment into the more informative labels *contradiction* and *unknown* (Giampiccolo et al., 2007). MacCartney and Manning (2009) proposed an extension of these label set to a set of seven so-called "basic semantic relations"; they are *equivalence*, *forward entailment*, *reverse entailment*, *negation* (corresponding to the *contradiction* label by Giampiccolo et al. (2007)), *alternation* (where two concepts are mutually exclusive, but are not complementary, such as *"cat"* and *"dog"*), *cover* (for concepts that have some overlap and cover the complete universe, such as *"animal"* and *"non-human"*), and *independence* for all other cases (examples taken from MacCartney and Manning (2009)). Dzikovska et al. (2013) use 5 labels in the SemEval-2013 task-7: *Correct* (corresponding to *entailment*), *Partially_correct_incomplete*, *Contradictory*, *Irrelevant* and *Non-domain*. Their label set is the first to consider entailment relations between answers in the context of questions; they do that by introducing the label *Non-domain* for answers that do not address the question.

We consider none of these label sets directly fitting for our task. In the classic RTE task (Dagan and Glickman, 2004; Giampiccolo et al., 2007), it is only of interest whether the *text* entails the *hypothesis* and not the other way round. In our annotation scenario, however, both

directions of entailment occur and are of interest to us. For naming conventions, we consider the learner answer to be the *text* and the target answer to be the *hypothesis*, such that an *entailment* relation means that the learner answer entails the target answer. We follow MacCartney and Manning (2009) and use the label *reverse entailment* to indicate entailment in the opposite direction, where the target answer entails the learner answer. This of course leads to the problem that a sentence pair consisting of two paraphrases would be ambiguous for annotation. Therefore, and in order to explicitly mark the equivalent cases, we also introduce *paraphrase* as a separate label, such that *entailment* and *reverse entailment* can only hold in the absence of a paraphrase relation, similar to the *equivalence*, *entailment*, and *reverse entailment* labels by MacCartney and Manning (2009).

We introduce the additional new label *partial entailment* to denote cases where some conceptual overlap between two answers is present, but the other entailment labels cannot be applied. Intuitively speaking, this label always applies when the learner got something about the answer right and no other entailment relation holds. This is typically the case when (1) parts of the required information are present in the learner answer and additionally some unnecessary information is added; or (2) when the learner answer and target answer are very similar without being in a entailment relation. This is the case, e.g., for close co-hyponyms, such as *"kaputt (broken)"* vs. *"verbogen (crooked)"* stated in an answer pair about a doll's pram. Case (1) closely corresponds to the notion of partial entailment described by Nielsen et al. (2009) where target answers are split into individual *facets*, small semantic units, that are either addressed by the learner or not. In their terminology, a partially entailing answer contains some of the required facets and in addition unnecessary ones. Note that for case (2), the question is important in order to determine whether two answers are semantically close. Consider a hypothetical question asking *"What did John eat?"* and a second one asking *"What type of cheese did John eat?"*. Consider additionally the target answer *"John ate swiss cheese"* and the learner answer *"John ate cheddar"*. In the context of the first question, the two answers are in a partial entailment relationship: The learner got right, that John ate some sort of cheese. For the second question, the information that John ate some type of cheese is already given in the question, so that the two answers do not have partial overlap. We would label them as *topical non-entailment* instead, a label we will introduce in the following paragraph.

So far, we have covered cases where at least a partial entailment relation between learner answer and target answer holds. The remaining cases, which do not have any overlap between learner answer and target answer, would be labeled as *unknown* in RTE or *independence* by MacCartney and Manning (2009). We further split them to accommodate the nature of our task of annotating answer pairs in the context of a question. Similar to Dzikovska et al. (2013) we take into consideration whether an answer addresses the corresponding question, i.e., whether it

Dagan et al.	entailment		non-entailment				-
Giampiccolo et al.	entailment		contradiction	unknown			-
Dzikowska et al.	correct		contradiction	partially correct incomplete		irrelevant	non-domain
MacCartney	equivalence	forward entailment	negation	alternation – cover – independence			-
our approach	paraphrase	entailment	contradiction	partial entailment	reverse entailment	topical non-entailment	off-topic

Table 4.2: Correspondences between different RTE label sets

is on-topic: Those answers that stand in no other entailment relation with the target answer but are still on-topic, receive the label *topical non-entailment*, while answers that do not address the right question are labeled as *off-topic*. This label is also similar to the notion of incongruence introduced by von Stechow (1990).

We now present examples and specifications for each of the labels used in our study:

paraphrase: Target answer and learner answer are paraphrases, i.e., express the same semantic content.

(4.2) **Question:** Wie viel verdient BA im Monat?

How much does BA earn monthly?

Target answer: BA verdient weniger als 300 Euro im Monat.

BA earns less than 300 Euro monthly.

Learner answer: weniger als 300 im Monat.

less than 300 Euro monthly

entailment: The learner answer textually entails the target answer, i.e., it is more specific than the target answer.

(4.3) **Question:** Außer den Sehenswürdigkeiten im ersten Absatz, was kann man in Dresden noch machen?

What can you do in Dresden apart from the sightseeing mentioned in the first paragraph?

Target answer: Man kann an der Uferprommenande einen Spziergang haben. (sic!)

You can take a walk by the waterfront.

Learner answer: An der Uferpromenade kann man einen erholsamen Spaziergang genießen.

You can enjoy a relaxing walk by the waterfront.

reverse entailment: The target answer textually entails the learner answer.

(4.4) **Question:** Woher kam der Heiligenschein?

Where did the halo originate from?

Target answer: Der Heiligenschein kam von dem Licht aus dem Ofen.

The halo originated from the light out of the oven.

Learner answer: Es kam aus der Ofen.

It originated from the oven.

partial entailment: There is a semantic overlap between target answer and learner answer but there is no clear entailment relation in any direction.

(4.5) **Question:** Nennen Sie zwei Orte, wo man draussen sitzen kann.

List two places where one can sit outside!

Target answer: Es gibt zwei große Terrassen und einen sonnigen Garten.

There are two large terraces and a sunny garden.

Learner answer: In den Garten oder Waldgebiet.

In the garden or forest area.

contradiction: Learner answer and target answer are mutually exclusive, i.e., they cannot both be true at the same time.

(4.6) **Question:** Ist die Wohnung in einem Neubau oder einem Altbau?

Is the apartment located in a new or an old building?

Target answer: Die Wohnung ist in einem Neubau.

The apartment is in a new building.

Learner answer: Die Wohnung ist in einem Altbau.

The apartment is in an old building.

topical non-entailment: The learner answer is in principle a valid (though not necessarily correct) answer to the question (it is *on-topic*) but there is no semantic overlap to the target answer that would qualify it for one of the other entailment categories.

(4.7) **Question:** Was war das Thema der Umfrage (survey)?

What was the topic of the survey?

Target answer: worauf könnte man nicht verzichten

things you can't do without.

Learner answer: Das Thema war Internet Nutzen.

The topic was usage of the Internet.

off-topic: While answers with any of the previous labels addressed the right question, i.e., were

on-topic, for this label, the learner answer is *off-topic*, i.e., it either answers a different question or it is a non-answer (such as *"I don't know"*) and therefore cannot be compared to the target answer.

(4.8) **Question:** Wer machte den Gartenzwerg berühmt?
Who made garden gnomes famous?
Target answer: Philipp Griebel machte den Gartenzwerg berühmt.
Philipp Griebel made garden gnomes famous.
Learner answer: Er war im thuringischen berühmt.
It was famous in the Thuringian.

In summary, our annotation schema accounts for both directions of the entailment relation by introducing the *reverse entailment* label in addition to *entailment*. It accommodates the properties of answer pairs (instead of sentence pairs) and the necessity to mark a partial entailment by splitting the *unknown* label by Giampiccolo et al. (2007) into three categories: *topical non-entailment*, *off-topic* and *partial entailment*. When comparing directly to the five diagnostic labels from the SRA data set (Dzikovska et al., 2013), we see a direct mapping only between their and our *contradictory* label, between *non-domain* and *off-topic* and between *irrelevant* and *topical non-entailment*. Their *correct* label subsumes both our *paraphrase* and *entailment* labels, thus implicitly assuming that a learner answer that entails the target answer is always correct. Their *Partially correct incomplete* also corresponds to two of our labels, *reverse entailment* and *partial entailment*. The difference between these two labels in our data set is that *reverse entailment* only lacks some information, while *partial entailment* additionally adds some unnecessary and potentially wrong information. From the diagnostic view, *reverse* and *partial entailment* have in common that something is missing in those answers and that they are therefore assumed to be wrong by (Dzikovska et al., 2013), a view that we wish to challenge.

Annotation Guidelines Our annotation guidelines cover the annotation scheme presented above. In addition, we introduce some specific guidelines due to the differences between classical RTE settings and the task and data we use, namely that of dealing with learner language and of labeling answer pairs.

Learner Language Issues: One feature of the data that makes the annotation in general difficult is the fact that learner answers in CREG are written by foreign language learners and therefore often contain spelling errors, come in an ungrammatical form or use lexically inappropriate material. Similar to what teachers do in a short-answer grading task, our annotators were instructed to ignore such errors as best as they could and base their decision just on the semantic content of an answer. That means they had to implicitly build a so-called *target hypothesis* for each

learner answer, i.e., an error-free grammatical version of what the learner presumably wanted to express (cf. Ellis (1994)), a task which is known to be problematic even for experienced teachers (Lüdeling, 2008).

This implies that the chosen label depends strongly on the interpretation of the annotator. For some answers, several interpretations with potentially also different RTE relations to the target answer are possible, as is illustrated by the following example:

(4.9) **Question:** Wo und wann fand man die meisten Gartenzwerge?

Where and when could most garden gnomes be found?

Target answer: Die meisten Gartenzwerge fand man in der Nachkriegszeit in Westdeutschland.

Most garden gnomes could be found in the postwar period in West Germany.

Learner answer hypotheses: Die Gartenzwerge setzte aus den Wald.

a) The garden gnomes [were] released in[to] the woods.

b) The garden gnomes set [something] out of the woods.

c) The garden gnomes marooned the woods.

The learner answer in this example is ungrammatical and could either be interpreted as *"The garden gnomes [were] released into the woods"'* (*"Die Gartenzwerge wurden im Wald ausgesetzt"*) or *"The garden gnomes put [something] out of the woods"* (*"Die Gartenzwerge setzten etwas aus dem Wald heraus"*) or *"The garden gnomes marooned the woods"'* (*"Die Gartenzwerge setzen den Wald aus"*), leading to *topical non-entailment* as the most plausible label for the first (a) and *off-topic* for the second (b) and third (c) interpretation. The reading text sentence *"Eine "Befreiungsfront für Gartenzwerge" stahl sie aus den Vorgärten und setze sie im Wald aus."* (*"A "garden gnome liberation front" stole them from the front gardens and released them into the woods."*) makes clear that the learner most likely and also erroneously lifted material from this text region, and indicates that the first interpretation is definitely to be preferred. However, we did not give the annotators access to the text on purpose, in order to prevent them from deciding whether an answer was correct or not. Therefore, the annotators were not in the position to make this reasoning. Note that the label *contradiction* is not an option for this particular answer: Although the question presupposes that there is only one correct answer and the topical reading a) of the learner answer gives a different location than the target answer, the two locations *"West Germany"* and *"in the forest"* are not mutually exclusive, but the learner answer rather addresses a different type of location than the target answer. A clear case of a contradictory answer is instead the following learner answer: *"Most garden gnomes could be found between 1948 and 1952 in the GDR"*, as GDR refers to a different location than West Germany.

Annotating Answers in Relation to the Question: In contrast to other RTE data sets that compare two texts, we display the answer pair together with a question they both address. The annotators can use the question in two ways: (1) to resolve anaphoric expressions such as pronouns occurring in the answers, and (2) to expand *term answers* in the form of ellipses to *full answers* (Krifka, 2001; von Stechow and Zimmermann, 1984).

Anaphora resolution occurs in the data mainly in the form of pronoun resolution; pronouns in an answer can often be resolved to an antecedent in the question. We see this phenomenon in the following example, where the pronoun *"sie"* (*"she"*) has to be resolved to *"Julchen"* from the question:

(4.10) **Question:** Warum kam Julchen in die Küche?
Why did Julchen come into the kitchen?
Learner answer (correct): Weil sie den Lärm horte.
Because she heard the noise.

For the phenomenon of ellipsis consider the following example:

(4.11) **Question:** Welche Stadt ist auf Platz eins? Warum?
What city is in first place? Why
Learner answer: Aachen. Man benutzt nur umweltfreundliches Papier.
Aachen. Only eco-friendly paper is used.

In this example, using information from the question is necessary in order to expand the term answer *"Aachen"* to the full answer *"Aachen ist auf Platz eins"* and to understand that there is an implicit causal discourse relation between the two sentences: *"Aachen ist auf Platz eins, weil man nur umweltfreundliches Papier benutzt."* (*"Aachen is in first place because they use only eco-friendly paper."*) The annotators were instructed to treat short and full answers in the same way. Specifically, only semantic content which has not been introduced by the question should be taken into consideration when deciding between the two labels of partial entailment and topical-non-entailment. We required this to avoid that a learner answer is considered partially entailed by the target answer if it is on-topic and repeats material from the question, but has otherwise no conceptual overlap with the target answer.

Semantic material introduced by the question is explicitly addressed in a *full answer* and omitted in a *term answer* in the terminology of, e.g., Krifka (2001), following von Stechow and Zimmermann (1984).

Annotation Process

All material has been double-annotated by two German native speakers with a background in computational linguistics using the *WebAnno* annotation tool (Yimam et al., 2013). The annotators were shown the question together with each learner answer-target answer pair, but could not see the corresponding text and did not know whether a learner answer had been graded as correct or incorrect. We did so to avoid that they would explicitly or implicitly base their labeling decision on the knowledge of whether an answer is correct or supported by the text. Cases of disagreement were additionally annotated by a third annotator and then resolved through majority voting. Instances where all three annotators gave a different label were resolved manually by another annotator, a co-author of this study (Ostermann et al., 2015).

4.1.2 Evaluation

The RTE annotations are now analyzed and compared to the two types of ASAS scores present in the CREG corpus: the binary correctness labels and more fine-grained five-way diagnosis labels.

Agreement

For the task of labeling entailment relations, annotators reached a Cohen's kappa Cohen (1968) of *0.69* which – according to Landis and Koch (1977) – indicates *substantial agreement*. The confusion matrix is given in Table 4.3. The labels *paraphrase*, *entailment*, and *reverse entailment* can be reliably identified by the annotators. However, the confusion matrix highlights two problems: first, the identification of *partial entailment* is not trivial, as can be seen from a relatively high rate of confusions between *partial entailment* and almost any other label, marked in dark gray in the table. Second, it seems to be hard to tell apart the three entailment classes *contradiction*, *off-topic*, and *topical non-entailment* (highlighted in light gray). These labels – as we will later see – primarily belong to answers scored as incorrect. When these labels are collapsed, the kappa score improves subsequently to *0.78*.

Additionally, we show in Table 4.4 the kappa values for Krippendorff's diagnostic (Krippendorff) in order to assess the difficulty of annotating each individual class. In this setup, all categories except the one under consideration are merged into one pseudo-class, and the inter-annotator agreement for this binary decision is measured. We can see here as well that annotation for partial entailment is most problematic, followed by topical non-entailment and contradiction.

	paraphrase	entailment	reverse entailment	partial entailment	contradiction	topical non-entailment	off-topic
paraphrase	**180**	4	12	9	0	2	0
entailment	6	**78**	0	15	0	2	0
reverse entailment	7	5	**112**	28	2	1	3
partial entailment	5	8	15	**75**	8	3	10
contradiction	0	0	0	2	**47**	1	1
topical non-entailment	1	0	2	10	35	**100**	30
off-topic	0	1	3	3	5	31	**169**

Table 4.3: Confusion matrix between the two annotators for entailment labels.

label	kappa
paraphrase	0.86
entailment	0.77
off-topic	0.74
reverse entailment	0.70
contradiction	0.61
topical non-entailment	0.56
partial entailment	0.50

Table 4.4: Krippendorff's diagnostic for label distinction, ordered by score

Comparison of Teacher Scores and Entailment Labels

In order to address the question of whether there is a direct mapping between RTE and ASAS labels, we compare our annotation to both teacher-assigned label sets given in the CREG corpus, the binary correctness and the five-way diagnosis labels.

Figure 4.5 shows the distribution of binary correctness labels in CREG over entailment classes. Some of the entailment labels clearly correspond to correct (*paraphrase*, *entailment*) or incorrect answers (*contradiction*, *off-topic*, *topical-non-entailment*). From the definition of these labels, this is an expected result: whenever a learner answer is a paraphrase of a target answer or more specific than a target answer it should be correct; whenever a learner answer contradicts the target answer, does not answer the question or answers the question without overlap with the target answer, it is most likely incorrect.

However, the labels *partial entailment* and *reverse entailment* cannot be as easily mapped to

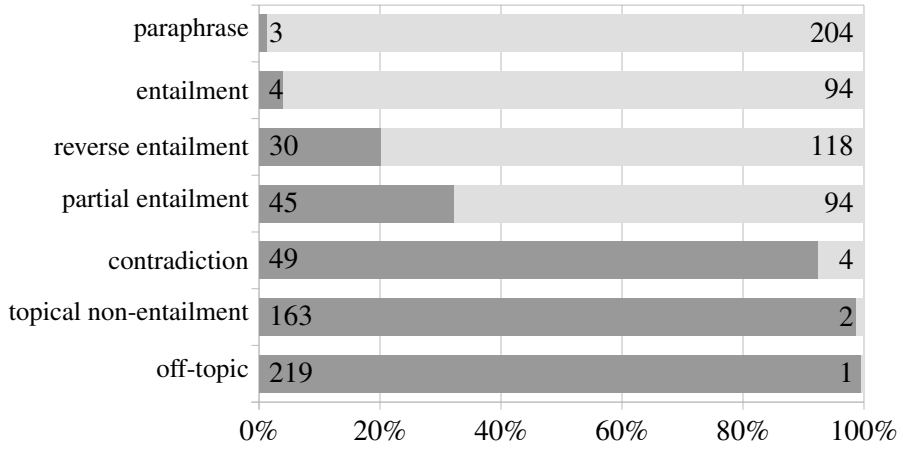

Figure 4.5: Distribution of binary labels over entailment classes (absolute values, correct: light grey, incorrect: dark grey).

binary scores, which indicates substantial differences between the two tasks of RTE and SAS. While one would expect from an RTE perspective that most answers labeled as *partial* or *reverse entailment* would be scored as incorrect by a teacher, the opposite is the case: 74% of all answers in those two categories are indeed marked as correct.

The two labels *partial* and *reverse entailment* have in common that only some information from the target answer is entailed by the learner answer (while in *partial entailment* the learner answer additionally entails information not present in the target answer). One possible explanation why such answers are often still scored as correct is that it seems that target answers are formulated in an exhaustive way and more elaborate than the teacher would expect the learner to answer. In other words, they are not minimal correct answers, that contain all information required for a good answer but not more. It might frequently be the case that the teacher mentions facts in their target answer that do not necessarily have to occur in a learner answer to make it correct. However, the target answer will certainly contain other key facts that have to occur. Unfortunately, it is not clear from the target answer which facts are necessary to make the learner answer correct and which are not.

Upon closer inspection, most of these correct answers can be grouped into one out of a few categories with respect to the material from the target answer that is not addressed in the learner answer (and for some answers several of the following phenomena are present):

List questions The intended redundancy of target answers becomes especially obvious in cases where a teacher asks a question such as *"Nennen Sie zwei Zimmer im Erdgeschoss*

(Name two rooms on the ground floor)" and lists more than two rooms in their target answer (*"Im Erdgeschoss gibt es ein Bad, Gäste WC, eine Küche und ein Wohn/Esszimmer"*). In other cases it is unclear whether the teacher asks for a single concept or a list, because they do not specify how many concepts the learner should mention. In such cases, the target answer seems to list all possible correct answers, while the teachers apparently are satisfied with a subset of them. Example 4.12 shows one answer pair for such a type of question, where the binary label is *correct*, although the entailment type clearly is *reverse entailment*. The target answer specifies a list of concepts out of which the learner answers only mentions one. Apparently, *"wood"* is the main concept that is considered necessary for a good answer, while *"water"* and *"energy"* might be optional. An answer just stating, e.g., *"water is needed"* does not occur in our corpus, but we consider it plausible that teachers label such an answer as incorrect due to the more prominent role of wood in the paper production process.

(4.12) **Question:** Was braucht man bei der Herstellung des Papiers?
What is needed for paper production?
Target answer: Bei der Herstellung des Papiers braucht man Holz, Wasser und Energie.
You need wood, water and energy to produce paper.
Learner answer: Man braucht Holz bei der Herstellung des Papiers.
Wood is needed for paper production.

Target answers containing additional modifiers not present in the learner answer

It also happens that a teacher specifies additional modifiers for a concept that are not necessary for the answer to be correct. These might be both restrictive modifiers, such as in example 4.13, as well as non-restrictive ones (*"die Tschernobyl-Katastrophe von 1986"* (*"the Chernobyl disaster of 1986"*) in the target answer vs. *"die Tschernobyl Katastrophe"* in the learner answer).

(4.13) **Question:** Was für Kenntnisse muss der Bewerber haben?
What types of skills does the applicant need?
Target answer: Er muss Sinn für Markt und Handel haben und praxis-gerechte Englischkenntnisse verfügen.
He needs to have a sense for markets and commerce and practice-oriented English skills.
Learner answer: Sinn für Markt und Handel, English.
sense for markets and commerce, English.

Target answers containing hypernyms of learner answer concepts In such cases, the

teacher again apparently over-specified the answer and a less specific concept such as *"man"* instead of *"soldier"* as in the following example is still considered precise enough. Again it is unclear what the boundaries for acceptable answers are. The question word *"wer"* already presupposes that a person is the expected answer, so, e.g., *"somebody"* would definitely not be precise enough.

(4.14) **Question:** Wer war an der Tür?
Who was at the door?
Target answer: Drei Soldaten waren an der Tür.
Three soldiers were at the door.
Learner answer: Die drei Männer war an den Tür.
The three men was at the door.

Off-topic content in the target answer In a few cases like the following example, the target answer contains additional material that does not address the question, like the temporal phrase *"seit sechs Jahren"*. Learner answers leaving out this information are still scored as correct.

(4.15) **Question:** Welchen Teil der Wirtschaft möchte die weißrussische Regierung fördern?
Which part of the economy does the Belorussian government want to promote?
Target answer: Die weißrussische Regierung fördert seit sechs Jahren den Ökotourismus.
The Belorussian government promotes eco-tourism since six years.
Learner answer: Der Okotourismus möchte die weißrussische Regierung förden.
The Belorussian government wants to promote eco-tourism.

In addition to these systematic cases of answers whose correctness label contradicts our expectations based on the RTE label, there are also a few curious cases of label-score combinations that seem completely implausible, such as answers with a negative entailment label that are scored as correct. The following example (4.16) illustrates this.

(4.16) **Question:** Wie lange hat die Firma schon eine Filiale in Frankfurt?
For how long does the company hold a branch at Frankfurt?
Target answer: Die Firma hat eine Filiale seit 15 Jahren in Frankfurt.
The company holds a branch at Frankfurt for 15 years.
Learner answer: Es hat eine Filiale in Erfurt für 15 Jahren.
It holds a branch at Erfurt for 15 years.

While our schema clearly labels the learner answer as *off-topic*, since question material is paraphrased in a wrong way (the question addresses a branch in Frankfurt, not Erfurt), the teacher decided to accept the answer by implicitly substituting the location of *"Erfurt"* with *"Frankfurt"*. This answer pair again highlights the problem of dealing with learner language: when a potential spelling mistake leads to another existing word like *"Erfurt"*, it is hard to decide whether the student really intended to write *"Erfurt"* on purpose, or whether it is a misspelling of *"Frankfurt"*, and thus a target hypothesis is hard to form. Given that the text does not mention Erfurt, it might have been plausible to a teacher to assume that the learner meant the right thing and to mark the answer as correct, while our annotators (without knowledge of the text) had no special reason to assume that *"Erfurt"* would be a spelling error.

Similarly, there are rare examples of *entailment* or *paraphrase* items that are labeled as *incorrect*. Example 4.17 shows one such pair, where, both for the entailment label and the correctness score, different options are plausible depending on the interpretation of *"warm light"* (temperature vs. color):

(4.17) **Question:** Warum legte der Mann das Holz in den Blechofen?
Why did the man put the wood into the plate oven?
Target answer: Er legte das Holz in den Blechofen, um das Zimmer wärmer zu machen.
He put the wood into the oven to make the room warmer.
Learner answer: Fuer ein warmes Licht durch das Zimmer.
For a warm light through the room.

In summary, the findings from this evaluation show that SAS and RTE are two separate tasks in our labeling scenario. This is in contrast to the work by Dzikovska et al. (2013) that assumes that the two tasks can be directly mapped to each other.

In addition to binary scores, the CREG corpus also contains a 5-way set of diagnosis scores. In these annotations, *missing concept* and *extra concept* are used if the answer missed important information or contained additional, unnecessary information, respectively. Therefore we would expect them to match our *reverse entailment* and *entailment* labels, while their *correct* label should correspond to our *paraphrase*. The label *blend* is a combination of *missing* and *extra concept*, seemingly similar to our *partial entailment*. One might expect that the label *non-answer* would correspond to our *off-topic*, however, it was indeed only used for real non-answers such as *"I don't know"*.
From the label descriptions, we would have expected to see a good fit between the two label sets, especially for the diagnosis labels *correct*, *missing concept* and *extra concept*. Instead, we

	correct	missing concept	extra concept	blend	non-answer
paraphrase	194	7	3	2	0
entailment	73	3	16	6	0
reverse entailment	74	53	1	20	0
partial entailment	50	37	8	46	0
contradiction	1	10	0	42	0
topical non-entailment	1	15	1	148	0
off-topic	1	40	0	175	4

Table 4.6: Confusion matrix for teacher assessment labels from CREG and entailment labels.

find that a clear mapping between our labels and the 5-way scores is not possible, as can be seen in the confusion matrix in Table 4.6. We assume that this is because teachers have a softer criterion for the definition of paraphrases. They mark items as correct – even if parts are missing or added, which leads to an RTE annotation of entailment or reverse entailment – but where the teachers felt that these omissions or additions were marginal or irrelevant for the correctness of the answer.

4.1.3 Conclusions

We presented in this section an annotation study that labels pairs of learner answers and target answers from the CREG corpus with a set of fine-grained textual entailment annotations which have been specifically tailored to accommodate annotating entailment relations between pairs of a question and corresponding answer. We tested the claimed correspondence between the two tasks of RTE and ASAS. Our main finding with respect to this research question is two-fold. First, there is a clear correspondence between some textual entailment classes and a binary correctness score. Learner answers that paraphrase or entail the target answer are almost always correct, while learner answers contradicting the target answers, irrelevant learner answers, and off-topic answers are almost always incorrect. Second, there is also an area that needs further investigation; this concerns the *partial* and *reverse entailment* cases, where – defying expectations – around 74% of the learner answers that fall into one of these two categories were marked as correct by teachers. These cases illustrate that the tasks of RTE and SAS are related but not equivalent for CREG. Parts of these differences come from multiple possible interpretations of a learner language answer, but in the majority of cases, teachers are indeed satisfied with an

answer that contains less or less specific material than their target answer.

These finding not only shed light on the relation between RTE and ASAS, but also provide insight on the grading behavior of teachers. It seems plausible to us that a teacher without any specific formal instructions sees a target answer as one instance of an example for a good answer and not necessarily as the minimal answer that contains all and only information necessary for a correct answer. We believe that teachers base their grading more on their intuition whether the student presumably understood the correct information, and that they score an answer as correct if it reasonably close to a correct answer.

In Section 5.3, we will further address the question of whether answers from those "problematic" classes of partial and reverse entailment are also challenging for an automatic scoring system and whether entailment labels can be predicted by standard ASAS features.

4.2 Annotation Study 2: Linking Answers to Text Passages

Students rely on textual material when answering reading comprehension questions: they reuse words, chunks or even whole sentences from the reading text verbatim for constructing their answers. This is especially true for language learners at early stages as they are likely to have a limited range of options both for lexical variance and grammatical constructions. The phenomenon is known as lifting and has been described as one strategy of language learners to answer reading comprehension questions (Anderson and Roit, 1996) and to produce language output in general (Carver, 1984). While the phenomenon of lifting has already been observed earlier for the CREG corpus (Ott et al., 2012), there have been no studies so far that addressed the relation between learner answers and the reading texts in detail.[1]

We fill this gap by annotating links between answer and sentences in the corresponding reading text for the example of the CREG corpus. We expect that most correct answers, both correct learner answers and (inherently correct) target answers, are paraphrases of a certain portion of the text, namely the part that answers the question, or are at least inferable from certain parts of the text. This makes intuitive sense since a text should contain the information necessary to answer the question, at least for those types of questions that lend themselves as good candidates for ASAS.

We have seen in Section 2.1 the question typology by Day and Park (2005), who distinguish six comprehension types: literal, reorganization, inference, prediction, evaluation, and personal response. We have argued there that only the first three types are good candidates for automatic

[1] We are restricting ourselves here to scenarios where a student is allowed to look at the text while answering the questions, in contrast to a variant where they first read the text and then answer questions while the text is not available anymore.

scoring, as the other three types elicit a wide variety of correct answers, which would make it hard to define a restricted number of target answers. Meurers et al. (2011c) annotated the comprehension type for the questions in the CREG corpus, following the typology of Day and Park (2005) and found that indeed only the first three occur in CREG: out of a total 177 questions, 140 (79.1 %) are literal questions, 24 (13.6 %) are reorganization questions and 13 (7.3 %) are inference questions.

As we assume lifting as a widely used strategy for dealing with reading comprehension questions, we hypothesize that wrong answers often contain textual material as well. But in contrast to correct answers, we assume that some of them originate from looking at the wrong portion of the text. In other words, one potential source of incorrect answers is an inability on the part of the student to correctly identify the portion of the text that is relevant to the question at hand. Our hypothesis therefore is that a learner answer which links to the same portion of the reading text as the target answer is likely to be a correct answer. Similarly, a learner answer which closely matches some part of the text that is *not* related to the target answer is likely to be incorrect. We expect the relation in the second hypothesis to be stronger, as identifying the correct region of the text is only the first step in finding the correct answer, and there are various ways that a student can still get the answer wrong after successfully identifying the correct sentence.

Our hypotheses are exemplified by Figure 4.7, which revisits the example from CREG with a reading text, a question about it, two target answers, one correct and two incorrect learner answers. We can see that both the target answers and the correct learner answer paraphrase the same sentence of the text (or parts of it); also the incorrect answers can each be clearly linked to a region of the text, but each to one that is different from the one containing the information for the correct answer.

Investigating the relation between learner answers and the text will not only provide insights into learner strategies when answering such questions, but might also be beneficial for the task of ASAS itself. Many approaches to short-answer scoring, especially those discussed under prompt-independent approaches in 2.4.2, only take the instructor-supplied target answer(s) and the question as their basis for scoring. The target answer is meant to indicate the semantic content necessary for a correct student answer. Alignment between student answer and target answer is then taken as a way of approximating semantic equivalence. However, a target answer is in fact just one way of expressing the requisite semantic content. Teachers who create such exercises are obviously also looking at the text while creating target answers, and target answers – just like learner answers – are often paraphrases of one or more sentences of the reading text. Therefore, we incorporate in Section 5.2 the reading text into the evaluation of the student answers by, e.g., a baseline classifier that simply checks whether learner answer and corresponding target answer refer to the same text sentence.

TEXT: SCHLOSS PILLNITZ
Das Schloss, das im Osten Dresdens liegt, ist für mich das schönste Schloss in Dresdens Umgebung. (...) Eine besondere Attraktion im Park ist die Kamelie . Die mittlerweile über 230 Jahre alte und 8,90 m hohe Kamelie bekam 1992 ein fahrbares Haus, in dem Temperatur, Belüftung, Luftfeuchte und Beschattung durch einen Klimacomputer geregelt werden. In der warmen Jahreszeit wird das Haus neben die Kamelie gerollt. Während der Blütezeit von Mitte Februar bis April trägt sie zehntausende karminrote Blüten. Ableger der Pillnitzer Kamelie werden jedes Jahr in begrenzter Zahl während der Blütezeit verkauft, dann ist ein Besuch besonders lohnend.

This palace, which lies in the east of Dresden, is to me the most beautiful palace in the Dresden area. (...) One special attraction in the park is the camellia tree. In 1992, the camellia, which is more than 230 years old and 8.90 meters tall, got a new, movable home, in which temperature, ventilation, humidity, and shade are controlled by a climate regulation computer. In the warm seasons, the house is rolled away from the tree. During the Blossom Time, from the middle of February until April, the camellia has tens of thousands of crimson red blossoms. Every year, a limited number of shoots from the Pillnitz camellia are sold during the Blossom Time, making it an especially worthwhile time to visit.

QUESTION:
Ein Freund von dir möchte sich die alte Kamelienpflanze ansehen. Wann sollte er nach Pillnitz gehen und warum gerade in dieser Zeit?
A friend of yours would like to see the historic camellia tree. When should he go to Pillnitz, and why exactly at this time?

TARGET ANSWERS:

- Von Mitte Februar bis April ist die Blütezeit.
  From the middle of February until April is the Blossom Time.
- Im Frühling trägt die Kamelienpflanze zehntausende karminrote Blüten.
  In spring the camellia has tens of thousands of crimson red blossoms.

LEARNER ANSWERS:

- [correct] Er sollte Mitte Februar bis April gehen, weil die alte Kamelienpflanze zehntausende karminrote Blüten trägt.
  He should go from the middle of February until April, because then the historic camellia has tens of thousands of crimson red blossoms.
- [incorrect] Der Pillnitzer Kamelie werden jedes Jahr in begrenzter Zahl während der Blütezeit verkauft.
  Every year, a limited number of Pillnitz camellia are sold during the Blossom Time.
- [incorrect] Alles Jahr wegen dem Temperatur und Luftfeuchte durch einen Klimacomputer geregelt werden.
  All year round against temperature and humidity are controlled by a climate regulation computer.

Figure 4.7: Example of reading text with question and answers (repeated from Figure 3.4. Links from an answer into the text are marked by using the same color for the answer and the corresponding source sentence.

4.2.1 Goal of the Study

The goal of this annotation study is to provide *source sentences* for answers in the CREG corpus, i.e., to annotate the sentence(s) a student or teacher most likely looked at when formulating answers to a question. We do this in order to understand the role of the text in answering short answer questions better and ultimately to average this understanding for the task of ASAS.

More precisely, we want to answer questions that are linked to research question 1.1:

RQ 1.1: How can the reading text be used in ASAS for reading comprehension tasks?

In detail, we address the following questions:

- How often is some part or all of an answer – either verbatim or a paraphrase of it – present in the text and if so, how often can it be linked to exactly one sentence?
- How reliably can humans determine those source sentences for an answer?
- Is there a correlation between the comprehension type (literal, reorganization, inference) and the difficulty of finding a source sentence, e.g., is the annotation easiest for literal questions and hardest for inference questions?
- Do correct learner answers link to the same sentence as the respective target answer (given the intuition that they should contain the same semantic content)?
- How often do incorrect learner answers link to a sentence in the text that is not the best sentence for the target answer, i.e., how often is an answer wrong because the student looked at the wrong part of the text?

With the first two questions, we address the difficulty of the annotation process depending on properties of the answers: whether it is a target answer, correct learner answer or incorrect learner answer and what the comprehension type of the respective question is. We do that by assessing agreement between two annotators. In order to address the last two questions about the linked region for learner answers and their corresponding target answers, we create an adjudicated gold standard as the basis for our analyses.

4.2.2 Contributions

This section makes the following contributions:

- We provide an additional annotation layer for the CREG-1032 corpus that specifies, for every answer that can be linked to an associated text, one best source sentence and potentially several additional source sentences.

- We analyze the inter-annotator agreement and find that the task is easier for target answers and correct learner answers than for incorrect learner answers, and easier for answers to questions of the literal comprehension type than for reorganization or inference questions.
- We confirm the hypothesis that correct learner answers indeed almost always link to the same text region as the corresponding target answer. Incorrect answers often, but not always, link to a different text passage in the reading text than the corresponding target answer.

4.2.3 Data and Annotation Process

As in Study 1, we also use the CREG corpus (Ott et al., 2012) for this annotation study and we also use the balanced subset CREG-1032 (Meurers et al., 2011a), see Section 3.2 for an overview of the corpus. One reading text with four corresponding questions, 10 target and 14 learner answers was not available for copyright reasons. Those answers have been excluded from annotation. For a better comparability of the results, we decided to split the reading texts into fixed units instead of asking the annotator to mark text passages of arbitrary length. Sentences are an obvious and conceptually intuitive unit with sentence splitting tools readily available, so we chose sentences as units. The reading texts have been automatically segmented into sentences using the OpenNLP Sentence Detector (Foundation, 2010).

For each answer to be annotated, annotators were given the reading text and the question and asked to first read the text and then identify *the single best source sentence from the text for that answer*. Annotators were not told whether any given instance was a target or a learner answer, nor whether learner answers were correct or incorrect.

Although we expected most answers to correspond directly to a single text passage, as most of them were of the literal comprehension type, annotators were asked to look for (and annotate appropriately) two different conditions in which more than one source sentence may be relevant. We refer to these as the repeated content condition and the distributed content condition. In the *repeated content condition*, the same semantic content may be fully represented in more than one sentence from the original text. In such cases, we would expect the text to contain sentences that are paraphrases or near-paraphrases of one another. The *distributed content condition* occurs when the relevant semantic content spans multiple sentences and some degree of synthesis or even inference may be required to arrive at the answer. Annotators were instructed to assume that pronouns had been resolved; in other words, a sentence should not be considered necessary semantic content simply because it contains the antecedent to which a pronoun in another sentence refers. For both of these multiple-sentence conditions, annotators were asked to select one single best source sentence and also to mark the alternative source sentences.

We had every answer annotated by two out of a total of three annotators (students of computational linguistics) such that each annotator annotated two thirds of the complete material. Each annotator saw approximately the same number of answers per text; they had to annotate all answers for one text before moving on to the next text and, within a text, all answers for one question before moving to the next question. The order of texts as well as the order of questions for each text and the order of answers for each question were randomized.

Following the first round of annotations, we discovered some instances that needed re-annotation. These were annotations that had been missing in the original annotation data set and had to be re-annotated afterwards. This could be either because they were not included in the original annotation set due to problems with wrong text ids or because one or both annotators from the first round did not provide any annotations at all and did not indicate in the comments that they worked on the item but could not identify a sentence. In such cases, we had two additional annotators label this answer.

Whenever the two annotators for an answer disagreed on the best sentence for an answer and there was no overlap in their alternatives, we requested an additional annotation from a third annotator.

4.2.4 Annotation Results

We collected a total of 2814 annotations for 1307 individual answers (1016 learner and 291 target answers). For 25.3 % of all annotations, the annotator provided more than one possible source sentence, the majority of which (638 out of 712) where annotated as distributed content. Upon closer inspection, though, the annotations for the two types of additional material are not very consistent. Due to these inconsistencies, we decided for the purpose of this study to treat the multiple-sentence conditions in an underspecified fashion and to consider the sentences in both conditions to be alternatives.

For assessing the difficulty of the annotation task, we identify four distinct categories with respect to agreement between two annotators. Note that this categorization does not include adjudication.[2]

agree: In this case, both annotators linked the answer to the same source sentence. That sentence is identified as the gold-standard link to the answer.

altagree: This category covers two different situations in which the two annotators fail to agree on the single-best sentence. First, there are cases in which the best sentence selected by

[2]Computing kappa values (Cohen, 1968) as a measure for inter-annotator agreement is often done when assessing the quality of annotations; but this metric is not directly applicable in this study: The number of categorial labels varies between texts, and per text the number of annotations is too small to derive reliable kappa values.

Answer type	agree	altagree	disagree	no link
leaner answers – all	74.9 %	10.5 %	12.9 %	1.8 %
learner answers – correct	79.2 %	12.2 %	8.6 %	0.0 %
learner answers – incorrect	70.3 %	8.7 %	17.3 %	3.7 %
target answers	78.2 %	9.8 %	12.0 %	0.0 %

Table 4.8: Inter-annotator agreement for linking answers to source sentences in the text

one annotator is in the set of alternatives indicated by the other. Second, in a small number of cases, there is no overlap between the best sentence for one and the alternatives for the other annotator, but both annotators agree on one or more alternatives. In other words, the intersection of the sets of sentences identified by the two annotators is taken as the gold-standard annotation. When the intersection contained more than one sentence, we only took the one occurring earliest in the text.

disagree: This category also includes two different types of cases. In the first, one of the annotators failed to identify a source sentence to link with the answer. In that case, we consider the annotators to be in disagreement. In the second case, the annotators disagree on the single-best sentence and there is no overlap between indicated alternative sentences.

no link: For a small number of answers, none of the annotators found a link to the text. For these cases, the gold standard provides no source sentence.

Table 4.8 lists the frequency of each agreement label for the subgroups of all learner answers, correct learner answers, incorrect learner answers and target answers, respectively. We can see from the table that it is indeed most often possible to reliably identify one best sentence for an answer (second column *agree*). This task is easier for correct learner answers and target answers and harder for incorrect learner answers. This is also reflected by a higher percentage of disagreements for incorrect answers. Answers for which no link could be found are very infrequent (n=18) and are exclusively incorrect learner answers.

Figure 4.9 gives an example for an incorrect learner answer for which no annotation could be found. In this example we see an instance where the learner apparently tried to answer the question based on their own common knowledge. Other instances of answers that cannot be linked to the text include non-answers such as the answer *"Ich weiß nicht. Sie muss erleiden, weil...?"* (*"I don't know, she has to suffer because ..."*) addressing a why-question by just parroting the given question material. Some questions involving inferencing were also hard to

TEXT: PAPIERATLAS
(. . .)Recyclingpapier spart Energie. Umweltschützer fordern (demand) außerdem, dass man Recyclingpapier benutzt. Recyclingpapier aus altem Papier ist besser für die Umwelt. Für die Herstellung braucht man nämlich weniger Holz und auch weniger Wasser und Energie. Schon mit drei Blatt (sheets) Recyclingpapier spart man so viel Energie, wie nötig ist, um eine Kanne (pot) Kaffee zu kochen.(. . .)

(. . .) Recycled Paper saves energy. Environmentalists also demand that you use recycled paper. Recycled paper made from old paper is better for the environment. This is because you need less wood and also less water and energy for the production. By using only three sheets of recycled paper you already save the energy necessary to cook a pot of coffee. (. . .)

QUESTION:
Was braucht man bei der Herstellung des Papiers?
What is needed for paper production?

TARGET ANSWER:
Bei der Herstellung des Papiers braucht man Holz, Wasser und Energie.
For paper production you need wood, water and energy.

INCORRECT LEARNER ANSWER:
Der Herstellung sollte ein Sammelstelle für Papier haben.
The production should have a collecting place for paper.

Figure 4.9: Example of an incorrect learner answer that could not be linked to the text.

link and either not linked at all or by just one annotator, such as the question *"Warum waren die Häuser kaputt?"* (*"Why were the houses broken down?"*) for the short story *"Die drei dunklen Könige"* by Wolfgang Borchert, where one has to infer from the whole setting that the story is placed directly after the Second World War.

Gold-Standard Creation The annotations were processed to produce a gold-standard set of source sentences, indicating for each answer the single sentence in the reading text to which it is most closely linked. In the *agree* case this was simply the sentence annotated by both annotators as best sentence. In the *altagree* case, we select the best sentence that was part of the alternatives of the other sentence or – in the case of only an overlap in the annotated alternatives, the intersection between the alternatives. If this intersection contained more than one sentence, we selected the sentences occurring first in the text as the gold annotation.

Cases of disagreement between the two annotators that could be resolved through a majority vote by a third annotator received this majority annotation as the gold annotation. In cases of disagreements after adjudication, we again selected the sentences occurring earliest in the text as the gold annotation.

comprehension type	agree	altagree	disagree	no link
literal	82.3 %	10.8 %	5.1 %	0.7 %
reorganization	75.0 %	12.9 %	3.0 %	0.9 %
inference	67.9 %	8.6 %	13.6 %	4.9 %

Table 4.10: Inter-annotator agreement for linking answers to source sentences in the text depending on the comprehension type of the question

correctness	goldlink	all annotators	alternatives
correct	78.3 %	91.9 %	98.3 %
incorrect	22.8 %	32.2 %	41.2 %

Table 4.11: Frequency with which both correct and incorrect learner answer link to the same sentence as the corresponding target answer in three different linking conditions.

4.2.5 Analysis: Correlation between Agreement and Comprehension Type

We expect that the process of linking to the text is easier for answers to questions that ask for literal information from the text. Therefore, we also compare the agreement labels for each of the three comprehension types *literal*, *reorganization*, and *inference*. In Table 4.10, we can see that the task is easiest for literal and hardest for inference questions. This confirms that questions that ask for verbatim information from the text are easier to locate than than those where information has to be collected or inferred.

4.2.6 Analysis: Do Correct Learner Answers always Link to the Same Sentence as Target Answers and Incorrect Answers to Different Sentences?

In a next step, we address the question of whether correct answers link to the same region of the text as the corresponding target answers and whether incorrect answers (at least often) stem from looking at the wrong portion of the text.

Table 4.11 shows the results under three slightly different conditions: One where we just take the annotated one best sentence for both learner and target answer for comparison, and check whether they are the same (goldlink), one where we consider the best source sentence for all annotators before adjudication (all annotators) and one where we additionally consider also

those sentences listed as alternatives by the annotators (alternatives). In such cases, we counted learner answer and target answer as linking to the same sentence if there is at least one sentence in the intersection of the sets for learner answer and target answer.

We can see that our intuition mostly holds, but is far from perfect. Correct learner answers indeed link most of the time to the same sentence as the target answer, especially in the relaxed alternatives condition. Figure 4.12 shows one rare example of a correct answer where target answer and learner answer do not have any source sentence alternatives in common. On closer inspection, this annotation is problematic. The text marked in blue, to which the target answer refers, contains an anaphoric pronoun *"mir"* that resolves to Bastian, not Heidi. However, we would not consider this to be a faulty annotation: A learner who writes such an answer might very likely indeed have looked at this sentence for spotting keywords from the question and finding an answer there, ignorant of the actual referent of *"mir"*. This question might be an example where the learner is on purpose led on the wrong track by phrasing the question in words actually occurring in the text but not describing the content relevant for the answer.

For incorrect answers, this correlation is less strict. Our annotations can be used to identify incorrect answers with a high precision but a low recall; 22% of all incorrect answers indeed link to the same text sentence as the corresponding target answer in the *goldlink* condition. This follows the intuition that there is more to a correct answer than identifying the right sentence. Figure 4.13 gives an example for such a case. While the student identifies the correct sentence, they rely on the wrong clause for their answer. This might be an indicator that our decision to annotate on sentence level might have been too coarse for some instances. Such an example could also be an indicator for a general problem with reading comprehension questions identified by Anderson and Roit (1996). They find that, especially when students lift material for their answers, it is not clear whether they actually understood the question or the material they used to form their answers; it could also very well be that they just mapped words in the answer to the text to find the presumably right sentences and to extract an answer from that text region, as most likely happened in the current example in Figure 4.13. However, in general, we see our hypotheses confirmed.

4.2.7 Conclusion

We have seen that both correct learner answers and target answers can be linked reliably to specific sentences in the reading text. This also holds (to a slightly lesser extent) for incorrect learner answers. These observations are especially true for questions of the literal comprehension type and to a lesser degree for inference and reorganization questions, which makes sense: those comprehension types require the student to either infer information that is not explicitly present in an answer or to combine information from several regions of the text. We have seen

TEXT: INTERVIEW MIT DREI DEUTSCHEN ÜBER IHRE BERUFSWÜNSCHE
(. . .) Bastian: Eigentlich möchte ich gerne im Freien arbeiten, (. . .).
Großes Ansehen zu haben oder viel Geld zu verdienen das ist mir nicht wichtig.
Til: Und du, Heidi? Welche Stelle würde dich in einem anderen Leben interessieren, wenn du die Wahl hättest?
Heidi: Am liebsten würde ich in meinem Beruf viel zu Hause sein und vielleicht sogar neue Rezepte (recipes) ausprobieren. (. . .)

Bastian: Actually, I would like to work outdoors, (. . .).
Having a good reputation or earning much money - that is not important for me.
Til: How about you, Heidi? What kind of job would interest you in a different life if you had the choice?
Heidi: I would prefer to stay a lot at home in my job and maybe even try out new recipes. (. . .)

QUESTION:
Sind Ansehen und Geld für Heidi wichtig oder unwichtig?
Are a good reputation and money important for Heidi or not?

TARGET ANSWER:
Für Heidi sind Ansehen und Geld unwichtig.
A good reputation and money are not important for Heidi.

CORRECT LEARNER ANSWER:
nicht wichtig - viel zu Hause sein, Rezepte ausprobieren.
not important - stay at home a lot, try out recipes.

Figure 4.12: Example of a *correct* learner answer that links to a *different* text sentence than the corresponding target answer.

TEXT: LUST AUF EINE STÄDTEREISE NACH DRESDEN?

(. . .) Dresden bietet sich für einen Kurzurlaub besonders an.
Wegen seiner italienisch inspirierten Architektur Elb-Florenz genannt, wurde die Metropole gegen Ende des Zweiten Weltkrieges von alliierten Bombern komplett zerstört.
Die Stadt hat alle Gebäude wieder aufgebaut, aber sich nie vollständig von diesem Schock erholt. (. . .)
(. . .) Dresden is especially suitable for a short trip.
Called "Florence on the Elbe river" because of its italian-inspired architecture, the metropolis was completely destroyed by allied bombers at the end of the Second World War.
The city restored all the building, but has never completely recovered from this shock.

QUESTION:
Welchen Einfluss hatte der Zweite Weltkrieg auf Dresden?
What was the influence the Second World War had on Dresden?

TARGET ANSWER:
Im Zweiten Weltkrieg war Dresden von alliierten Bombern komplett zerstört.
In the Second World War, Dresden was completely destroyed by allied bombers.

INCORRECT LEARNER ANSWER:
Der Zweite Weltkrieg hatte Einfluss aus Italien.
The Second World War had influence from Italy.

Figure 4.13: Example of an incorrect learner answer that links to the same text sentence as the corresponding target answer.

that incorrect learner answers might – but don't necessarily have to – result from looking for the answer in the wrong place in the text. While correct learner answers almost always link to the same region in the text as the respective target answers, incorrect answers often do as well, i.e., the student extracts some wrong information from the right passage. We will see later in Section 5.2 that the source sentence for a learner answer can be determined automatically and that our observations can be used for a simple baseline classifier that just checks whether learner answer and target answer link to the same text. We can imagine that such a check is also potentially helpful in giving feedback, e.g., in a tutoring system: if a student identified the wrong part of the text as containing the answer to the question, a tutoring system could inform her about it and suggest to look somewhere else or even indicate the region where to look for an answer.

4.3 Annotation Study 3: Textual Entailment Relations between Answers and Text Passages

This study addresses the entailment relations both between text passages in a reading text and learner answers as well as text passages and target answers.

In Study 1 of this chapter, we have examined the textual entailment relations between learner answers and their corresponding target answers, based on the idea that a correct learner answer entails its corresponding target answer, while an incorrect answer does not.

For the task of reading comprehension, as present in the CREG corpus, we have further shown in Study 2 that almost all answers can be linked to source sentences that specify the textual material used to build an answer. This also confirms our expectation that the answers to reading comprehension questions of the types present in CREG (literal, reorganization and inference questions) can indeed be found in the reading text. While the aim of Study 2 was simply to identify links between answers and passages of the text, Study 3 aims to type those links using entailment labels.

In terms of RTE, we hypothesize that the reading text also entails correct answers. In the case of incorrect answers, two cases are plausible: (a) The information in an incorrect answer is not or only partially entailed by the text, or (b) the text entails this information, but it is not the right answer to the question.

Consider the (re-visited) example 4.14 that shows part of a reading text, a question about the text, the target answer and two learner answers (one correct, one incorrect). We can see that both the target answer and the correct learner answer follow from the text: the bold-print passage of the text entails it. Furthermore, this learner answer is also entailed by the target answer. In this case, they are not paraphrases but the learner answer contains some additional information. The incorrect learner answer is also entailed by the text, but by a sentence that is different from the

TEXT: SCHLOSS PILLNITZ
Das Schloss, das im Osten Dresdens liegt, ist für mich das schönste Schloss in Dresdens Umgebung. (...) Eine besondere Attraktion im Park ist die Kamelie. Die mittlerweile über 230 Jahre alte und 8,90 m hohe Kamelie bekam 1992 ein fahrbares Haus, in dem Temperatur, Belüftung, Luftfeuchte und Beschattung durch einen Klimacomputer geregelt werden. In der warmen Jahreszeit wird das Haus neben die Kamelie gerollt. **Während der Blütezeit von Mitte Februar bis April trägt sie zehntausende karminrote Blüten. Ableger der Pillnitzer Kamelie werden jedes Jahr in begrenzter Zahl während der Blütezeit verkauft, dann ist ein Besuch besonders lohnend.**

This palace, which lies in the east of Dresden, is to me the most beautiful palace in the Dresden area. (...) One special attraction in the park is the camellia tree. In 1992, the camellia, which is more than 230 years old and 8.90 meters tall, got a new, movable home, in which temperature, ventilation, humidity, and shade are controlled by a climate regulation computer. In the warm seasons, the house is rolled away from the tree. **During the Blossom Time, from the middle of February until April, the camellia has tens of thousands of crimson red blossoms. Every year, a limited number of shoots from the Pillnitz camellia are sold during the Blossom Time, making it an especially worthwhile time to visit.**

QUESTION:
Ein Freund von dir möchte sich die alte Kamelienpflanze ansehen. Wann sollte er nach Pillnitz gehen und warum gerade in dieser Zeit?
A friend of yours would like to see the historic camellia tree. When should he go to Pillnitz, and why exactly at this time?

TARGET ANSWERS:

- Von Mitte Februar bis April ist die Blütezeit.
  From the middle of February until April is the Blossom Time.

LEARNER ANSWERS:

- [correct] Er sollte Mitte Februar bis April gehen, weil die alte Kamelienpflanze zehntausende karminrote Blüten trägt.
  He should go from the middle of February until April, because then the historic camellia has tens of thousands of crimson red blossoms.
- [incorrect] Alles Jahr wegen dem Temperatur und Luftfeuchte durch einen Klimacomputer geregelt werden.
  All year round against temperature and humidity are controlled by a climate regulation computer.

Figure 4.14: Example of a reading text, question, and answers from CREG.

one that the target answer is linked to. The incorrect learner answer does not answer the question and stands in no entailment relation with the target answer.

Goal of this Study

In this study, we look at the entailment relations between both learner answers and target answers and the reading text.

This study addresses both research questions 1.1 and 1.2:

RQ 1.1: How can the reading text be used in ASAS for reading comprehension tasks?

RQ 1.2: Is there a direct mapping between the two tasks of RTE and ASAS?

We do this by looking at analogies between RTE and ASAS again, as in Study 1, but with a focus on text-answer pairs, instead of pairs of learner and target answers. More precisely, we address the following questions.

- What entailment relations hold between relevant parts of the reading text and learner as well as target answers in CREG?
- Are these relations different for correct (learner and target) answers in comparison to incorrect learner answers?

Contributions

This study makes the following contributions:

- We provide entailment annotations between answers in CREG and the corresponding sentence of the text that were provided as source sentences by Study 2.
- Our main finding is that the largest part of answers is supported at least partially by the text, independent of their correctness. We also find that there is a rather clear correspondence between the degree of text support and the correctness of an answer, where correct answers are in general better supported by the text.

4.3.1 Data and Annotations

As for the other two studies in this chapter, we use the CREG-1032 corpus. We use our previous annotations from Study 2, where we annotated source sentences for each learner answer, as basis for our annotations. We presented annotators (native speakers of German with a background in computational linguistics) with pairs consisting of an answer (either a learner answer or a target answer) and one or several source sentences. They labeled each pair with one of the following

six labels: *paraphrase*, *entailment*, *reverse entailment*, *partial entailment*, *contradiction*, and *independence*. The label definitions correspond to the ones in Study 1 with the exception that *independence* replaces the labels *topical non-entailment* and *off-topic*. This is due to the fact that in contrast to Study 1, we do not consider whether an answer addresses the question or not, but we are only interested in whether the propositional content of an answer is entailed by the text or vice versa. This also changes the role of the question in an answer: annotators were asked to use the question to resolve pronouns and to expand term answers, but the question focus does not play a role anymore since the label *topical non-entailment* is not used in this study. Also, in contrast to Study 1, the information structure in the question does not play a role anymore, so it does not matter which information is already given or not.

As in all of our annotation studies in this chapter, annotators had to deal with learner language when annotating pairs containing a learner answer and were asked to annotate based on their understanding of the text; i.e., they were asked to form a target hypothesis (but were not asked to write it down).

4.3.2 Annotation Results

We obtained double annotations for 964 learner answers/text pairs and 255 target answers/text pairs. Our annotators reached an inter-annotator agreement of 0.53 on the 6-way classification task indicating moderate agreement (Landis and Koch, 1977), showing that the task is more difficult than in Study 1. Cases of disagreement have been resolved by a third annotator.

The differences between the two labels *paraphrase* and *entailment* is often not meaningful for the purpose of study: in both cases, the answer is entailed by the text; the concrete usage of a label just depends on whether the answer agrees with the sentence boundaries of the text passage or not. Learner answers annotated as *paraphrase* indeed contain a high number of cases where a learner lifted a whole sentence from the text, so that one might argue that a learner just copied a sentence without really understanding it. A similar argumentation holds for *partial* and *reverse entailment*: while for reverse entailment, the learner answer contains all information from the text passage plus some more specific or additional information, for partial entailment, the learner answer might contain only information from part of the passage. This again is an indicator that our decision to annotate whole sentences in Study 2, instead of, e.g., clauses, might have been suboptimal and a more fine-grained annotation would provide further insights.

For these reasons and in order to simplify the presentation of results in the following analyses, we collapse the labels *paraphrase* and *entailment* to *full support*, *reverse entailment* and *partial entailment* to *partial support* and *contradiction*, and *independence* to *no support*. This class merge did, however, not influence inter-annotator agreement.

Comparing these annotations with the binary correctness annotations that come with CREG,

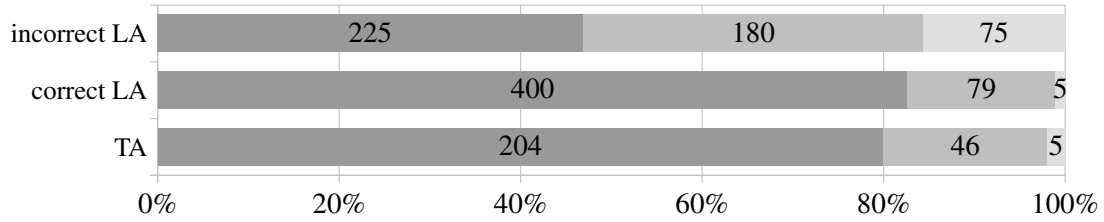

Figure 4.15: Amount of text support for incorrect learner answers, correct learner answers and target answers. Levels of support are, from left to right, *full*, *partial* and *no support*.

we find that correct answers are much better supported by the text than incorrect answers. Target answers have a similarly good support from the text as correct learner answers. Curiously, there are some target answers and correct learner answers which have only partial support and a few with no support. These considers some answers where the context is not clearly represented in a particular text sentence, but where some text sentence was nevertheless identified in Annotation Study 2 as source sentence. Consider the following example.

(4.18) **Question:** Warum waren die Häuser kaputt?
Why were the houses broken?
Target answer: Die Häuser waren kaputt wegen des Krieges.
They were broken because of the war.
Learner Answer: Die Häuser waren kaputt, wegen Krieg..
They were broken because war.
Supporting Text Passage: In drei alten Uniformen.
In three old uniforms.

In this case the very short text sentence annotated as source sentence in Annotation Study 2 alone does clearly not support the answer. However, it was considered as a source sentence in Annotation Study 2, probably because the mention of uniforms was the most obvious pointer for the annotator how the learner could identify that the short story was set during or directly after a war.

Even among the incorrect learner answers, many have full support of the text, showing that students lifted their answers from the text but selected the wrong material. The following example shows such an incorrect answer that is fully supported by the text (and is labeled as *off-topic* in Study 1), but incorrect as the student lifted the wrong material:

(4.19) **Question:** Was hatte die Nachbarin vorher getan? Warum?

What did the neighbor do before and why?

Target answer: Sie hatte in ihrem Verschlag gekramt, um ein Weihnachtsgeschenk für Julchen zu finden.

She rummaged around in her compartment to find a Christmas present for Julchen.

Learner Answer: Die Nachbarin hatte ein Geschenk für Julchen.

She had a Christmas present for Julchen.

Supporting Text Passage: Ich habe etwas für Julchen zu Weihnachten, sagte sie.

I have something for Julchen for Christmas, she said.

Other cases of incorrect answers that have at least partial support from the text are those where the learner fell for a distractor item, i.e., their answer addresses the question. Such an answer is typically labeled in Study 1 as *topical non-entailment* or *contradiction*. Consider the following example:

(4.20) **Question:** Wie nannte man die Gartenzwerge zuerst?

What were garden gnomes called at first?

Target Answer: Man nannte die Gartenzwerge ”Gnome”.

Garden gnomes were called ”gnomes”.

Supporting Text Passage for Target Answer: Anfangs hießen die Zwerge noch ”Gnome” und ihre Hersteller ”Gnömchenmacher.”

In the beginning, the dwarfs wer called “Gnome” and their manufacturers ”Gnömchenmacher.”

Learner Answer: Zuerst nannte man die Gartenzwerge Tarrkotta-Zwerge.

First the garden gnomes were called clay gnomes.

Supporting Text Passage for Learner Answer: Manche halten ihn für ca. 130 Jahre alt, weil Philipp Griebel im thüringischen Gräfenroda schon 1872 einen Terrakotta-Zwerg machte und in seinen Garten stellte.

Some think it is 130 years old because Phillip Griebel already made a garden gnome in the Thuringian Gräfenroda in 1872 and placed it in his garden.

In this example, the learner gave an answer that might in theory be a good answer to the question (i.e., it is on topic). Both target answer and learner answer contain NPs as the new information in the answer that occur in the text and are used to refer to a garden gnome, but the learner selected the wrong one.

4.3.3 Conclusions

In this study, we provide annotations about the entailment relation between reading text passages and learner and target answers. We observe that both correct learner answers and target answers typically have full text support, i.e., are entailed, by the reading text. Incorrect learner answers are not completely or not at all supported by the text in more than 50 % of all answers. This again supports our hypothesis in Study 2 that incorrect learner answers can still be linked to the text but that the learner either extracts their answers from the wrong passage or in an incomplete way, such that an entailment relation to the text no longer holds.

4.4 Conclusions

The main findings from the three studies are the following: (a) It became clear from Annotation Study 1 that the tasks of ASAS and RTE are not equivalent and that a direct mapping between RTE labels for the relation between a learner answer and the corresponding target answer and binary correct/incorrect scores assigned to the learner answer by teachers is not possible. (b) In Study 2, we showed that almost all learner and target answers (with the exception of a few incorrect learner answers, mostly non-answers) can be linked to one or more source sentences in the corresponding reading text. Even incorrect answers are most of the time grounded in the text. While correct learner answers almost always refer to the same text passage as their corresponding target answer, incorrect answers frequently (but not always) do not. This study confirms that learners lift material from the text for their answers and that one source of error comes from selecting the wrong text passage to extract the answer. (c) In Study 3, we labeled the entailment relation between answers and source sentences and found that correct (learner and target) answers are often entailed by the text, while incorrect learner answers often have at least partial support from the text.

If we combine the labels from Study 1 and Study 3, we observe that the classes *paraphrase* and *entailment* have the highest proportion of answers that are entailed by the text. This, of course, correlates to the fact that they also have the highest proportions of correct answers. At the same time, *topical non-entailment*, *off-topic* and especially *contradiction* have the highest proportion of answers not entailed by the text, again correlated with the high proportion of incorrect answers.

The combination of RTE labels with the target answer and with the text allows us to make assumptions about the correctness of that answer: e.g., paraphrases that are entailed by the text should be correct, and topical non-entailment answers not entailed by the text are very likely incorrect.

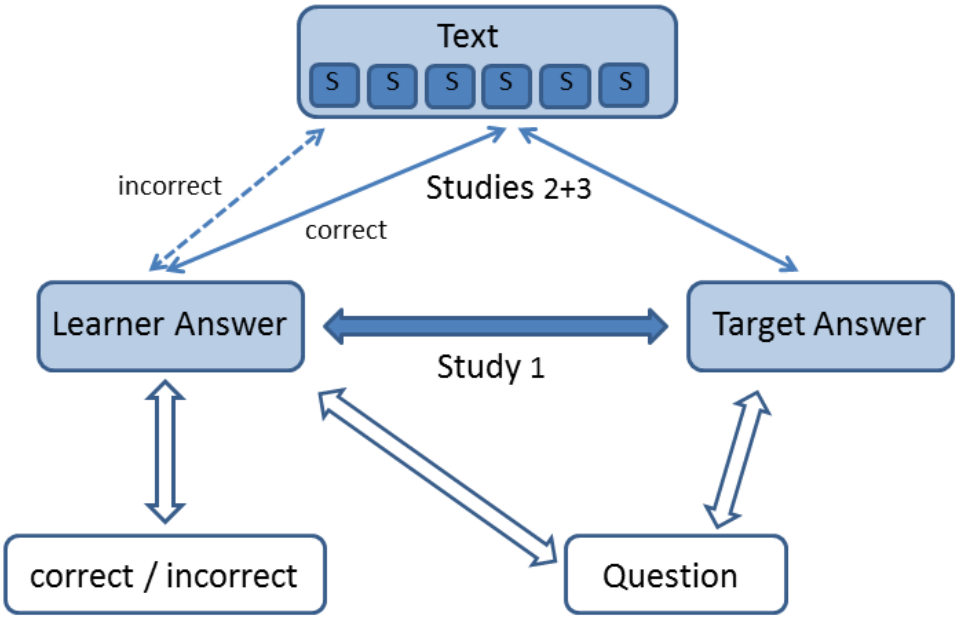

Figure 4.16: Visualization of the relations targeted in the three annotation studies.

We also observe some unexpected label combinations. Some combinations of correctness RTE labels from Study 1 and Study 3 should intuitively not happen, such as cases where an incorrect answer has full text support and is not labeled as *off-topic*. However, such cases occur frequently in our data (136 cases, half of which are labeled as *topical non-entailment*). However, it is debatable in most cases whether an answer is really on-topic or not, as demonstrated in the following example, where knowledge of the Austrian cuisine, while not being a formal job training, still might be considered as a relevant qualification for a restaurant job.

(4.21) **Question:** Was für eine Ausbildung braucht die Person, die beim Berliner Restaurant arbeiten will?

What kind of training does a person who wants to work at the Berlin restaurant need?

Target Answer: Die Person braucht eine abgeschlossene Berufsausbildung als Koch.

The person needs a finished apprenticeship as chef.

Learner Answer: Kenntnis in der Österreichischen Küche.

Knowledge of the Austrian cuisine.

Supporting Text Passage for Learner Answer: Wir erwarten [...] - Berufserfahrung und Kenntnisse in der österreichischen Küche.

> We expect [...] working experience and knowledge of the Austrian cuisine.

We use the findings from Study 1 and Study 2 in the following part of the thesis on ASAS on the CREG corpus (see Chapter 5): (a) We build a binary classifier that automatically determines a best sentence for a learner answer and checks whether this sentence is the same as the gold sentence from Study 2 for the target answer. (b) We integrate text-based features into an alignment-based scoring model that exploits alignments between learner and target answer (both in Section 5.2). (c) We use the RTE labels from Study 1 in several oracle-style machine learning evaluations (Section 5.3). We find that instances that are misclassified by an automatic classification model belong primarily to the RTE classes *partial* and *reverse entailment* that we identified as problematic for the mapping between RTE and ASAS. We further find that adding gold-standard RTE labels used as a feature improve classification accuracy for binary correctness labels and that an ASAS feature set can also be used to predict the RTE labels.

5 Experimental Studies – Automatic Short Answer Scoring

Our corpus studies in Chapter 4 highlighted the relation between learner answers and their connected materials – target answers and reading texts – by means of manual annotations. We build on these findings and conduct experiments on the *automatic scoring* of short answers in this chapter. We investigate approaches for fully automatic scoring using supervised machine learning techniques that include the relation of a learner answer to the target answer and to the reading text.

The studies in this chapter build on the corpus studies from the previous chapter and aim at integrating findings from these studies into our scoring models. We present the following individual studies:

- **Experimental Study 1** in Section 5.2 builds on Corpus Study 2 from Section 4.2. In that study, we found that most answers can be linked to one or more sentences in the reading text which the learner or teacher used to formulate their answer. We show that we can also reliably find this best sentence within a text automatically. We use this best sentence in a basic automatic classifier that simply checks whether a learner answer links to the same sentence as the corresponding target answer. We additionally investigate the use of text-specific features for the task of ASAS.

- **Experimental Study 2** in Section 5.3 uses gold-standard information from Corpus Study 1 in Section 4.1 where we annotated textual entailment relations between learner and target answers. We conduct experiments that investigate potential improvements if we could perfectly determine entailment relations between target and learner answers automatically.

- Our annotation studies confirmed that correct answers are indeed often paraphrases of the target answer and that also incorrect answers have parts (so-called fragments) that are paraphrasing parts of the target answer. **Experimental Study 3** in Section 5.4 aims at automatically detecting such paraphrase fragments. We leverage methods from statistical machine translation to learn statistical alignments between learner and target answers.

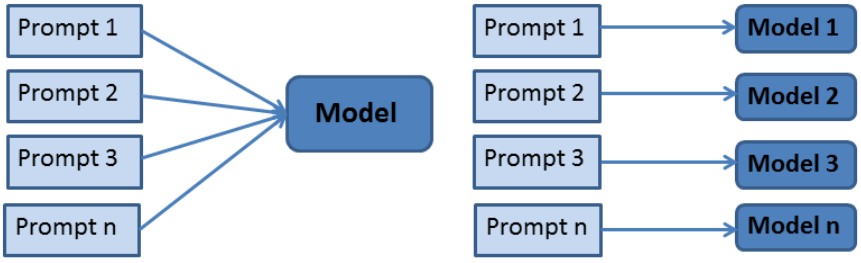

Table 5.1: Visualization of prompt-specific and prompt-independent models

Those alignments then provide the means to extract paraphrase fragments from the answer pairs. Properties of these fragments are in turn used as features for ASAS.

Relationship between experimental studies in this chapter and Chapter 6 This thesis presents two groups of experimental machine learning studies in this chapter and in the following one. While the experiments in this chapter are theoretic in nature, we bring the ASAS task closer to an application scenario (in Chapter 6) by a set of experimental studies, where we investigate how to reduce a teacher's annotation workload for ASAS, or in other words how to achieve an optimal scoring performance for a given restricted number of human annotation steps. We call this second set of experimental studies *assisted scoring*, because they model a scenario where a teacher receives support in scoring a set of unlabeled learner answers.

Note that both sets of experimental studies use – at least in part – supervised machine learning techniques, i.e., they both rely on the availability of annotated training data. In the automatic scoring studies (this chapter), we investigate different scoring techniques in general and use a fixed data set for that. In the assisted scoring studies, we focus on the number and selection of instances that have to be labeled by a teacher in order to reach a certain performance.

The methods presented in the automatic scoring studies are mainly *prompt-independent* due to the nature of the corpus used (CREG, see Section 3.2). In contrast, we use *prompt-specific scoring* in the assisted scoring studies. In prompt-independent scoring, one model is learned for all answers from a data set that includes several prompts; in prompt-specific scoring, a separate model is learned for each prompt (see Figure 5.1, see also Section 2.4 for this distinction).

The decision whether a prompt-specific model or a prompt-independent model is used depends on a number of factors: First, a necessary pre-condition to train a prompt-specific model is the availability of enough training material, i.e., enough learner answers per prompt. This is, e.g., not the case in the CREG-1032 subset corpus used for our automatic scoring studies

with 1032 learner answers that address 167 unique answers leading to on average 6.2 answers per question. Of course, the question how many training instances constitute "enough data" is debatable and depends both on the variety of answers that can be expected for a prompt and the lowest acceptable performance of the classifier in a certain scoring setting. In real-world applications such as exams or placement tests, it is often the case that a high number of answers to the same question has to be scored. We have these application scenarios in mind, when we conduct our experimental studies for assisted scoring.

Second, the type of model is dependent on the kind of features one wants to use, or, the other way round, the model choice influences what features can be used in a model. A prompt-specific model can work on lexical features, such as specific words or n-grams occurring in the learner answer. A prompt-independent model cannot do this. Prompt-independent models typically model the correctness of an answer by measuring the overlap with or similarity to some sort of target answer. That means the availability of a target answer is a pre-requisite for the applicability of prompt-specific models. This property is, e.g., not present in the ASAP data set, where only scoring guidelines but no examples for correct answers are given.[1]

5.1 Baseline: An Alignment-Based ASAS Approach

The work we present here is strongly based on approaches towards ASAS by Meurers and colleagues (Bailey and Meurers, 2008; Meurers et al., 2011a,c; Ziai et al., 2012), see also Section 2.4.2.

In order to compare to this previous work, we re-implement an alignment-based model following that proposed in (Meurers et al., 2011c). We refer to this class of models as *answer-based* because they function by aligning learner answers to target answers along several different dimensions, as discussed below. Learner answers are then classified as correct or incorrect on the basis of features derived from these alignments.

Wherever possible or practical, we directly re-implement the Meurers model. Here, we describe relevant aspects of the Meurers model, along with modifications and extensions in our implementation of that model.

[1] It would of course be possible to imagine a scoring setting where learner answers receiving a full score are used as target answer after some initial manual scoring.

Preprocessing

We preprocess all material (learner answers, target answers, questions and reading texts) using standard NLP tools for sentence splitting and tokenization (both OpenNLP[2]), POS tagging and stemming (both Treetagger (Schmid, 1994)), NP chunking (OpenNLP), and dependency parsing (Zurich Parser (Sennrich et al., 2009)). We use an NE Tagger (Faruqui and Padó, 2010) to annotate named entities. Synonyms and semantic types are extracted from GermaNet (Hamp and Feldweg, 1997).

Given that we are dealing with learner language, but do not want to penalize answers for typical learner errors, spellchecking (and subsequent correction of spelling errors) is especially important for this task. Our approach in this study is as follows: we first identify all words from the learner answers that are not accepted by a German spellchecker (aspell[3]). We then check for each word whether the word nevertheless occurs in the target answer, question or reading text. If so, we accept it as correct. Otherwise, we try to identify (using Levenshtein distance) which word from the target answer, question, or reading text is most likely to be the form intended by the student.[4]

Prior to alignment, we remove all punctuation, stop-words (restricted to determiners and auxiliaries), and material present in the question from the answer.

Alignment

The alignment process in short answer scoring approximates the determination of semantic equivalence between target answer and learner answer. During alignment, we identify matches between answer pairs on a number of linguistic levels: tokens, chunks (such as noun phrases), and dependency triples.

On the token level, we consider a number of different metrics for identity between tokens, with each metric associated with a certain alignment weight. After weights have been determined for all possible token pairs, the best applicable weight is used as input for a traditional marriage alignment algorithm (Gale and Shapley, 1962).

We use the following types of identity, weighted in the following order:

- token identity: two tokens are string-identical, e.g., *geht - geht*
- lemma identity: same lemma, but different word forms, e.g., *gehe - gehen*

[2] `http://opennlp.apache.org/index.html`
[3] `http://aspell.net/`
[4] In Chapter 7 we focus on the problem of dealing with learner language and propose a more informed way to handle spelling mistakes.

- spelling identity: the learner answer word has been spellchecked to the word in the target answer, e.g., *Sepisen - Speisen*.
- synonym identity: We take synonymy in a broader sense than just members of the same synset, but extract everything that is at most two levels (in both directions) apart as potential synonyms of a word.
- similarity identity: the two words have a GermaNet path relatedness above some threshold (taking the maximum over all possible synset pairs for two wordforms).
- semantic type identity: two words have the same GermaNet hypernym from a list of relevant hypernyms (e.g., animal, food, profession,...).
- NE identity: two NEs are the same, determined via heuristics on string matching, e.g., *Hans Schmidt - Herr Schmidt*, we are also linking NEs of type person to personal pronouns.
- NE type identity: two NEs are not the same but are referring to instances of the same type, e.g., two different locations.
- POS identity: For certain functional words, it might be reasonable to consider not only lemma-identical words for alignment, but also words of the same or related POS classes. Only some closed-class words are eligible for POS identity. We treat, e.g., all types of determiners as POS identical. By this type of identity we recognize that direct, indirect, and demonstrative articles could be possibly aligned, or that two prepositions or modal verbs could be considered alignable.

Unlike, for example, statistical alignment in machine translation, in which every token pair is considered a candidate for alignment, only candidates with at least one pre-specified type of identity are available for alignment under the Meurers model. This aims to prevent completely unrelated word pairs from being considered for alignment. We call this kind of alignment *knowledge-based alignment*. In Section 5.4, we consider statistical alignment as an alternative.

In order to favor alignment of content words over alignment of function words, and in departure from the original Meurers model, we use a content word multiplier for alignment weights that is applied to nouns, verbs and adjectives.

Chunks can only be aligned if at least one pair of tokens within the respective chunks has been aligned, and the percentage of aligned tokens between learner and target answer chunks is used as input for the alignment process. Dependency triple pairs are aligned when they share dependency relation, head lemma, and dependent lemma.

Features

After answers have been aligned, the following features are extracted as input for a machine learning classifier: target token overlap (percentage of aligned target tokens), learner token overlap (percentage of aligned learner tokens), token match (percentage of token alignments that are token identical), lemma match, synonym match, type match, target triple overlap, learner triple overlap, target chunk overlap, learner chunk overlap, target bigram overlap, learner bigram overlap, target trigram overlap, learner trigram overlap, variety of alignment (number of different token alignment types), and keyword overlap (percentage of aligned keywords, as keywords, we consider all nouns in the target answer).

The n-gram features are the only new features in our re-implementation of the Meurers model, hoping to capture the influence of linear ordering of aligned tokens. In the end, these features did not improve the model's performance.

5.2 Experimental Study 1: Using Links between Answers and the Reading Text for Automatic Scoring

Reading comprehension exercises are one particular type of short-answer questions and a common means of assessment for language teaching: students read a text and are then asked to answer questions about the text. Figure 5.2 shows an exemplary reading text, a question about it, the connected target answers, and a set of learner answers for that question, all from the CREG corpus. The nature of a reading comprehension task in the context of foreign language learning is that the student is asked to show that he or she has *understood* the text at hand and is able to extract information from it. Questions focus on one or more pieces of information from the text, and correct responses should contain the relevant semantic content. As we have discussed earlier, in Section 2.3, responses classified as correct might still contain grammatical or spelling errors; the focus in ASAS lies on the content rather than the form of the learner answer.

In Corpus Study 2 and 3 in Sections 4.2 and 4.3, we have found that both correct and incorrect answers can be linked by human annotators to one or more sentences in the text which contains the information used for constructing the answer. We have further shown that correct answers are most of the time entailed by those annotated text passages, while information in incorrect answers is often only partially supported by the text. Therefore, also the automatic scoring of short answer responses to reading comprehension questions can in essence be seen as a textual entailment task, with the additional complication that the learner must have identified the right portion of the text in order to answer a question correctly. It is not enough that a learner answer is entailed by *some* part of the reading text; it must be entailed by the part of the text which is

TEXT: SCHLOSS PILLNITZ

This palace, which lies in the east of Dresden, is to me the most beautiful palace in the Dresden area. (...) One special attraction in the park is the camellia tree. In 1992, the camellia, which is more than 230 years old and 8.90 meters tall, got a new, movable home, in which temperature, ventilation, humidity, and shade are controlled by a climate regulation computer. In the warm seasons, the house is rolled away from the tree. During the Blossom Time, from the middle of February until April, the camellia has tens of thousands of crimson red blossoms. Every year, a limited number of shoots from the Pillnitz camellia are sold during the Blossom Time, making it an especially worthwhile time to visit.

QUESTION:
A friend of yours would like to see the historic camellia tree. When should he go to Pillnitz, and why exactly at this time?

TARGET ANSWERS:

- From the middle of February until April is the Blossom Time.
- In spring the camellia has tens of thousands of crimson red blossoms.

LEARNER ANSWERS:

- [correct] He should go from the middle of February until April, because then the historic camellia has tens of thousands of crimson red blossoms.
- [incorrect] Every year, a limited number of Pillnitz camellia are sold during the Blossom Time.
- [incorrect] All year round against temperature and humidity are controlled by a climate regulation computer.

Figure 5.2: Example of reading text with question and answers from CREG

responsive to the question under discussion. (In Section 4.1, we have additionally investigated the resemblance between ASAS and RTE in terms of the relation between a learner answer and the corresponding target answer in that a correct leaner answer often entails the target answer, while an incorrect one does not.)

Previous approaches to automatic short answer scoring have seldom considered the reading text itself, instead comparing learner answers to target answers supplied by instructors; we will refer to these as *answer-based models*. In this study, we explore the role of the text for short answer scoring, evaluating several models for considering the text in automatic scoring on the German CREG corpus.

Goal of the Study

Like the Corpus Studies 2 and 3 in Section 4.2 and 4.3, this study addresses the research question 1.1:

RQ 1.1: How can the reading text be used in ASAS for reading comprehension tasks?

This study investigates how the annotated relations between answers and the text can be used for the task of ASAS. In detail, we pursue the following questions:

- How can we use the observation that correct learner answers most of the time link to the same text passage than the target answer while incorrect answers often do not?
- Can we improve classification performance of a standard alignment-based model by features extracted from the relation between the learner answer and the text?

Contributions

We show that the use of text-based features improves classification performance over purely answer-based models slightly (84% vs. 82% accuracy). We also find that a very simple text-based classifier, while it does not achieve the same performance as the answer-based classifier, does reach an accuracy of 76% for binary classification (correct/incorrect) of learner answers. The implication of this for automatic scoring is that reasonable results may be achievable with much less effort on the part of instructors; namely, a classifier trained on the supervision provided by marking the region of a text relevant to a given question performs reasonably well, though not as well as one trained on full target answers.

5.2.1 Baseline: Answer-Based Models

As a baseline, we use an alignment-model based on alignments with the target answer as described in Section 5.1. For classification, we use the timbl toolkit (Daelemans et al., 2009) for k-nearest neighbors classification following the original Meurers model. We treat all features as numeric values and evaluate performance via leave-one-out cross-validation. Further details appear in Section 5.2.3.

5.2.2 Text-Based Models

Previous prompt-independent approaches to ASAS on CREG take the instructor-supplied target answer(s) as a sort of supervision; the target answer is meant to indicate the semantic content necessary for a correct learner answer. Alignment between learner answer and target answer is then taken as a way of approximating semantic equivalence. The key innovation of the current

study is to incorporate the reading text into the evaluation of learner answers. In this section, we describe and evaluate three approaches to incorporating the text. The aim is to consider the semantic relationships between target answer, learner answer, and the text itself.

A target answer is in fact just one way of expressing the required semantic content. Teachers who create reading comprehension exercises are obviously looking at the text while creating both questions and target answers, and target answers are thus often paraphrases of one or more sentences of the reading text, as we have seen in Section 4.2. Some learner answers which are scored as incorrect by the answer-based system may in fact be variant expressions of the same semantic content as the target answer, where the scoring model is unable to detect this equivalence. Due to the nature of the reading comprehension task, in which students are able to view the text while answering questions, we expect students to express things in a manner similar to the text. This is especially true for language learners, as they are likely to have a limited range of options both for lexical expression and grammatical constructions.

Along similar lines, we have also seen in Study 1 in Section 4.2 that one potential source of incorrect answers is an inability on the part of the student to correctly identify the portion of the text that is relevant to the question at hand. We found that a learner answer which links to the same portion of the reading text as the target answer is likely to be a *correct* answer. Similarly, a learner answer which closely matches some part of the text that is *not* related to the target answer is likely to be *incorrect*.

Our text-based models investigate these finding with respect to ASAS considering three different models for incorporating the reading text into automatic short answer scoring. In the first approach, we employ a purely text-based model. A second approach combines text-based features with the answer-based baseline model, our third approach combined both the text-based and the answer-based model. Evaluation of all three approaches appears in Section 5.2.3.

To our knowledge, there is no previous work that uses reading texts as evidence for short answer scoring in the context of foreign language learning.

Simple Text-Based Model

This model classifies learner answers by comparing the source sentence most closely associated with the learner answer to that associated with the target answer. If the two sentences are identical, the answer is classified as *correct*, and otherwise as *incorrect*.

We consider both the annotated best sentences (*goldlink*) and automatically-identified answer-sentence pairs (*autolink*). For automatic identification, we use the alignment model described in Section 5.2.1 – originally meant to compare pairs of target answers – to identify the best matching source sentence in the text for both learner and target answers. We use the token alignment process of the alignment model to align a given answer with each sentence from its

respective reading text; the best-matching source sentence is that with the highest summed-up alignment weight. Chunk alignments are used only for correction of token alignments, and dependency alignments are not considered here. Thus an answer is matched to the sentence in the text with which it has the highest textual overlap.

This model takes an extremely simple approach to answer classification and could certainly be refined and improved. At the same time, its relatively strong performance (see Table 5.4) suggests that the minimal level of supervision offered by teachers simply marking the sentence of a text most relevant to a given reading comprehension question may be beneficial for automatic answer scoring.

In addition to the purely text-based model, we next explore two ways of combining text- and answer-based models.

Textual Features in the Answer-Based Model In the first model, we extract four features from the alignments between answers and source sentences and incorporate these as additional features in the answer-based model.

We compute two versions of those four features: One where we use as source sentences the annotated gold standard (*goldlink*) and one where we use the best matching sentence as they are found by our alignment model (*autolink*). The second feature is equally computed in both conditions.

1. **SourceAgree** This boolean feature is true if both learner and target answer link to the same source sentence, and false otherwise (also if no source sentence was annotated or automatically found).

2. **SourceEntropy** For this feature we look at the two most likely source sentences for the learner answer, as determined by automatic alignment scores. We treat the alignment weights as probabilities, normalizing so that they sum up to one. We then take the entropy between these two alignment weights as indicative of the confidence of the automatic alignment for the learner answer.

3. **AgreeEntropy** Here we weight the first feature according to the second, taking the entropy as a confidence score for the binary feature. Specifically, we value **SourceAgree** at 0.5 when the feature is true, -0.5 when false, and multiply this with $(1 - \text{entropy})$.

4. **TextAdjacency** This feature captures the distance (in number of sentences) between the source sentence linked to the learner answer and that linked to the target answer. With this feature we aim to capture the tendency of adjacent passages in a text to exhibit topical coherence (Mirkin et al., 2010).

model	k=5	k=15	k=30
baseline	0.817	0.820	0.822
baseline+syn	0.822	0.826	0.825
text: goldlink	0.827	0.827	0.829
text+syn:goldlink	0.830	0.835*	0.837*
text:autolink	0.837*	0.836*	0.825
text+syn:autolink	0.844*	0.836*	0.832
combined	0.810	0.819	0.816
combined+syn	0.817	0.822	0.818

Table 5.3: Classification accuracy for answer-based baseline (**baseline**), answer-based plus textual features (**text**), and classifier combination (**combined**). **+syn** indicates expanded synonymy features, **goldlink** indicates identifying the source sentences via annotated links, **autolink** indicates determining source sentences using the alignment model, k=number of neighbors. Results marked with * are significant compared to the best baseline model. See Section 5.2.3 for details.

Classifier Combination In the second approach, we combine the output of the answer-based and text-based classifiers to arrive at a final classification system, allowing the text-based classifier to dominate in those cases for which it is most confident and falling back to the answer-based classifier for other cases. Confidence of the text-based classifier is determined based on entropy of the two highest-scoring alignments between learner answer and source sentence. The entropy threshold was determined empirically to 0.5.

5.2.3 Experiments and Results

This section evaluates the answer-based and text-based models described above. In all cases, features and parameter settings were tuned on a development set which was extracted from the larger CREG corpus, such that there is no overlap between test and development data. For testing, we perform leave-one-out cross-validation on the subset of the corpus which was used in the first round of annotation containing 889 instances (see Section 4.2).

Answer-Based Baseline

As a baseline for our text-based models we take our implementation of the answer-based model from (Meurers et al., 2011c). As previously mentioned, our implementation diverges from theirs

in some points, and with 82.6% accuracy, we do not quite reach the performance reported for their model (accuracy of 84.6% on the balanced CREG corpus) and are far from reaching the current state-of-the-art accuracy of 86.3%, as reported in Hahn and Meurers (2012).

Our answer-based model appears as *baseline* in table 5.3. During development, the one extension to the baseline which helped most was the use of extended synonyms. This variant of the model appears in the results table with the annotation *+syn*.

Text-Based Models

As described in Section 5.2.2, we consider three different approaches for incorporating the reading text into answer classification: use of textual features in the answer-based model, combination of separate answer-based and text-based models, and a simple text-based classifier.

Adding Textual Features to the Answer-Based Model We evaluate the contribution of the four new text-based features, computed in two variations: with source sentences as they are identified in the gold standard (*goldlink*) and as they are computed using the alignment model (*autolink*). We add those additional features to the two answer-based systems: the baseline (*text*) and the baseline with extended synonym set (*text+syn*). Results are presented in table 5.3.

We present results for using the 5, 15, and 30 nearest neighbors for classification, as the influence of various features changes with the number of neighbors. We calculate the significance for the difference between the best baseline model (0.826) and each model which uses textual features, using a resampling test (Edgington, 1986). The results marked with a * in the table 5.3 are significant at $p \leq 0.01$.

Although the impact of the textual features is clearly not as big with a stronger baseline model, we still see a pattern of improved accuracy.

Additionally, it is surprising to see, that using goldlink information does not necessarily provide an advantage over autolink features, but both feature groups are in a similar range.

Classifier Combination Combining the two classifiers (answer-based and text-based) according to confidence levels results in decreased performance compared to the baseline. These results appear in table 5.3 as *combined*.

Simple Text-Based Classification We have seen that textual features improve classification accuracy over the answer-driven model, yet this approach still requires the supervision provided by teacher-supplied target answers. In our third model, we investigate how the system performs without this degree of supervision, considering how far we can get by using *only* the text.

	autolink	**goldlink**	**alt-set**
Accuracy	0.762	0.722	0.747
P correct	0.805	0.781	0.753
R correct	0.667	0.585	0.702
F correct	0.729	0.668	0.727
P incorrect	0.735	0.689	0.742
R incorrect	0.851	0.849	0.788
F incorrect	0.789	0.761	0.764

Table 5.4: Classification accuracy, precision, recall, and F-score for simple text-based classifier, under three different conditions.

The simple text-based classifier, rather than taking a feature-based approach to classification, bases its decision solely on whether or not the learner and target answers link to the same source sentence. We compare three different methods for obtaining these links. The first approach (*autolink*) automatically links each answer to a source sentence from the text, based on alignments as described in Section 5.2.1. The second (*goldlink*) uses links as provided by the gold standard; in this case, learner answers without a linked sentence (e.g., *nolink* cases) are immediately classified as incorrect. The third approach (*alt-set*) exploits that fact that in many cases annotators provided alternate source sentences. Under this approach, an answer is classified as correct provided that there is a non-empty intersection between the set of possible source sentences for the learner answer and that for the target answer. For the second and third approaches, we classify as incorrect those learner answers lacking a gold-standard annotation for the corresponding target answer.

In Table 5.4 we present classification accuracy, precision, recall, and F-score for the three different conditions. Precision, recall, and F-score are reported separately for correct and incorrect learner answers. The 76% accuracy reached using the simple text-based classifier suggests that a system which has teachers supply source sentences instead of target answers and then automatically aligns learner answers to the text, while nowhere near comparable to the state-of-the-art supervised system, still achieves a reasonably accurate classification. Again, it is interesting to see that the goldlink condition performs below the autolink condition, especially because of a higher Recall of correct / Precision of incorrect answers, where apparently the annotators in the gold standard failed to identify a good candidate for an answer.

5.2.4 Conclusions

In this study we have presented the first use of reading texts for automatic short-answer scoring in the context of foreign language learning. We show that, for CREG, the use of simple text-based features improves classification accuracy slightly over purely answer-based models. More importantly, we have also shown that a simple classification model based only on linking answers to source sentences in the text achieves a reasonable classification accuracy. These findings are strongly tied to the properties of the task: We are dealing with reading comprehension for language learners with limited linguistic expressiveness. Thus it seems plausible that we can leverage the link into the text in such a way. These findings also have the potential to reduce the amount of teacher supervision necessary for authoring short answer exercises within automatic answer scoring systems: It would be sufficient just to highlight the region in the text containing the information answering the question instead of formulating a target answer. In addition, our finding might be used in an exercise type where students are asked to identify the region containing the correct answer to a short-answer question. We assume that this exercise is easier than the full ASAS task and could thus be used either as a pre-stage to the actual ASAS task or as an independent exercise type.

5.3 Experimental Study 2: Using Textual Entailment Relations between Answers in Automatic Scoring

After we modeled the correctness of answers to short-answer questions using the text in the previous study, we will next have a look at whether information about entailment relations with the target answer can improve short answer scoring. We also investigate how scoring features can model entailment relations.

Goals of the Study

We addressed the relatedness of the two tasks of ASAS and RTE in Corpus Study 1 and 3 in Chapter 4 from an annotation perspective. In the present study, we want to further investigate this relationship through machine learning experiments.

This follow-up study to Corpus Studies 1 and 3 will help us to answer research question 1.2. from the machine learning viewpoint:

RQ 1.2: Is there a direct mapping between the two tasks of RTE and ASAS?

Our previous studies found that there is no one-to-one mapping between the two label sets. Instead, the entailment classes of *partial* and *reverse entailment* contain considerable numbers

of both correct and incorrect answers. To further assess the nature of the relation between the two tasks, we address here the following research questions:

- Can gold-standard information about the entailment relation between a learner answer and its corresponding target answer help the task of ASAS? We address this question in **Experiment 1** by adding gold-standard entailment labels as an additional feature.
- How well can we learn our annotated entailment classes using standard ASAS features? **Experiment 2** is meant to answer this question and also investigates whether ASAS features are indeed tailored towards detecting the correctness of an answer or whether they are implicitly more geared towards detecting textual entailment.
- State-of-the-art binary short answer scoring approaches label about 86% of the corpus correctly. In order to understand the challenges of automatic scoring better, we evaluate in **Experiment 3** which instances in terms of our entailment annotation labels are most problematic for automatic scoring with a binary label.

Contributions

In the experiments in this study we find that combining ASAS features with gold-standard RTE labels improves ASAS classification accuracy above the performance when using each feature set in isolation. When using ASAS features to predict RTE classes instead of correctness labels, performance goes down, indicating that current feature sets indeed address ASAS rather than RTE. When comparing the performance of ASAS on members of the different entailment classes, we find that instances of the controversial entailment classes *partial* and *reverse entailment* are also problematic for automatic scoring.

5.3.1 Data and Experimental Setup

We explore the relation between RTE and ASAS through a series of machine learning experiments. As with our previous studies, we use the CREG-1032 corpus. As our baseline ASAS feature set, we use our re-implementation of the Meurers et al. (2011c) *alignment* model, as described in Section 5.2, that reaches an accuracy of 86 %. (Note that differences to the accuracy reported in the section before are due to a different classification algorithm and due to the fact that we evaluate on the complete balanced data set, instead of the ca. 900 answers used in the previous study.)

For Experiment 1, we extend this feature set with the 7-way gold-standard entailment information from Corpus Study 1. In Experiment 2, we use these entailment labels as classes that we want to learn. In Experiment 3, we evaluate the performance of the baseline ASAS feature

set by entailment type. For all experiments, we use the *Logistic* classifier in the *Weka* package, that is based on a logistic regression algorithm (Hall et al., 2009), as we found this classifier to work better than the knn classifier used in the previous study. All experiments were evaluated via leave-one-out cross-validation.

Experiment	Feature set	Class label	Accuracy	Kappa
Experiment 1	alignment	correctness	*0.861*	*0.723*
	gold entailment	correctness	*0.905*	*0.827*
	alignment + gold entailment	correctness	*0.922*	*0.843*
Experiment 2	alignment	entailment-7	*0.473*	*0.36*
	alignment	entailment-5	*0.641*	*0.489*
	alignment	entailment-3	*0.749*	*0.562*
	alignment	entailment-2	*0.837*	*0.668*

Table 5.5: Experiment 1 and 2: Overview of the classifier performance for different learning tasks using various label and feature sets.

5.3.2 Experiment 1: Entailment Features for ASAS

In this experiment, we enhance the feature set used by the classifier with our annotated entailment label as an additional feature. Our goal is to explore whether our annotations would be helpful in the ASAS task, if they could be determined automatically. The additional feature raises the classification performance from 86.1% accuracy *($\kappa = 0.723$)* to 92.2% *($\kappa = 0.843$)*, as can be seen in Table 5.5. Although we showed that the RTE and ASAS scenario differ substantially, this outcome emphasizes that they also have a lot in common. If we consider entailment information as the only feature in an ASAS task, we reach an accuracy of 90.5%; this shows that there are contributions from both the alignment features and the entailment feature. Note that such a classifier with just this one entailment feature classifies all *paraphrase*, *entailment*, *reverse entailment*, and *partial entailment* classes as correct and all others as incorrect and yields a good recall for correct cases at the expense of precision. See also Figure 4.5 in Section 4.1 for the confusion matrix between entailment and correctness labels.

The usage of a manually annotated feature is of course comparable to the use of a human oracle and is therefore not feasible for a fully automatic approach. Thus, further research has to concentrate on how we can automatically model entailment types computationally. This leads to the question of to what extent the alignment-based baseline model is already able to predict our entailment types. An evaluation of this question is presented in the next experiment.

5.3.3 Experiment 2: Learning Entailment Relations

In the second experiment, we address the question of how well the automatic prediction of entailment labels is possible with the feature sets of an alignment based ASAS approach. Although the focus in an educational application is the automatic scoring of the correctness of a learner answer rather than its entailment relation to the corresponding target answer, this experiment sheds additional light on the relation between the two tasks.

We therefore train our classifier on the learner answer data using the 7-way entailment information as class label (*entailment-7* in Table 5.5). This leads to an accuracy of 47% and a kappa indicating *poor agreement*. The confusion matrix for this classification (Table 5.6) shows that the machine learner especially struggles with labeling the negative classes *contradiction*, *topical non-entailment*, and *off-topic*. This is because the features used are based on the alignment between target answer and learner answer, while the topicality of the answer is not modeled. Therefore, the machine learner is unable to decide if an answer addresses the question or not. *Partial entailment* poses a large difficulty as well, resulting in an F-Score of 0.336 *(*$P = 0.348$*/*$R = 0.324$*)* for that class. In contrast, the F-Score for *paraphrase* reaches a modest level of 0.649 *(*$P = 0.638$*/*$R = 0.66$*)*.

To narrow down the difficulties for our machine learner, we stepwise collapse our entailment labels, by first subsuming the negative entailment classes *topical non-entailment*, *off-topic* and *contradiction* as one class (*entailment-5*), which leads to only 5 entailment classes and an accuracy of 64.1%. In the next step, we subsume *entailment*, *reverse entailment* and *paraphrase* under one "positive" label, but leave partial entailment out, which leads to 3 classes (positive, negative, partial) and an accuracy of 74.9% (*entailment-3*). Finally, we add *partial entailment* to the positive class and achieved a performance of 83.7% (*entailment-2*). Although it is in general not surprising that the performance increases as the number of labels decreases, it is interesting that the inclusion or exclusion of *partial entailment* has a rather high impact on the performance. Overall we see that the performance on the 2-way entailment task is still slightly below the performance for binary correctness classification.

5.3.4 Experiment 3: Performance of Automatic Scoring by Entailment Type

ASAS approaches for CREG often rely on alignments based on both surface and conceptual overlap between learner answers and target answers, i.e., similar to our RTE annotations, they address which parts of a learner answer are represented in the target answer and vice versa. We have seen that the two entailment classes *partial* and *reverse entailment* contradict the expectation that they would contain mainly answers labeled by teachers as incorrect and instead contain

real \ classified	paraphrase	entailment	reverse entailment	partial entailment	contradiction	topical non-entailment	off-topic	recall
paraphrase	136	11	26	18	1	3	11	*0.66*
entailment	20	44	2	24	0	1	7	*0.45*
reverse entailment	24	1	82	17	0	1	23	*0.55*
partial entailment	20	15	20	46	0	9	32	*0.32*
contradiction	5	2	7	7	1	3	28	*0.02*
topical non-entailment	2	3	10	13	2	16	119	*0.10*
off-topic	6	3	14	7	1	26	163	*0.74*
precision	*0.64*	*0.56*	*0.51*	*0.35*	*0.20*	*0.27*	*0.43*	

Table 5.6: Experiment 3: Confusion matrix for the alignment model on entailment labels with precision and recall for all classes.

similar levels of both correct and incorrect answers. In this experiment, we want to address the question of whether instances of these two classes are also more problematic for automatic scoring. To do so, we apply the baseline alignment model and count how many instances of each entailment class are misclassified.

Figure 5.7 shows the distribution of these incorrectly classified instances over entailment types. We can see that learner answers labeled with *partial entailment* or *reverse entailment* are more problematic for the ASAS model than the other labels. An explanation is that an alignment-based ASAS model cannot know whether a token or phrase in the target answer that was not covered in the learner answer was crucial for a correct answer or not.

The machine learner also struggles in general with the *contradiction* class. This is because many contradicting answer pairs still provide a high lexical overlap but differ in just a small but critical detail as highlighted by the following example.

(5.1) **Question:** Ist der Text von einer Frau oder einem Mann geschrieben?

Is the text written by a woman or by a man?

Target answer: Der Text ist von einer Frau geschrieben.

The text is written by a woman.

Learner answer: Einem Mann hat der Text geschrieben.

A man wrote the text.

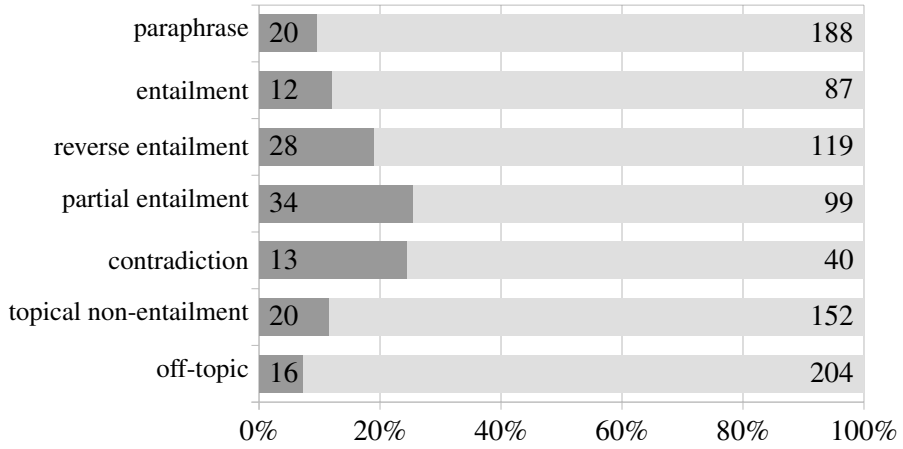

Figure 5.7: Correctly (light grey) and incorrectly (dark grey) classified instances per entailment class, relative and absolute values.

5.3.5 Conclusions

Our modeling experiments show that oracle entailment information as an additional feature improves classification performance substantially from 86.1% to 92.2% accuracy. This seems intuitively plausible, as entailment and correctness information are often highly correlated, as we have seen in Annotation Study 1. Knowing one kind of information is therefore very helpful in predicting the other one.

In the second experiment, we have seen that these entailment relations can be learned with existing feature sets reasonably well, but only up to an accuracy that lies below that for ASAS. These findings again highlight the relatedness of both tasks. The second experiment also confirms that standard ASAS models are indeed more suitable for the task they are meant for, i.e., detecting the correctness of an answer rather than for detecting entailment relations. This finding is a bit surprising, because we felt that the difference between positive and negative entailment classes was conceptually easy to understand, while it is sometimes not clear to us, when an answer that is not entailed by the learner answer is still ”good enough” to count as correct. We might assume that a model is not able to learn those slightly fuzzy categories.

The third experiment has shown that those answers that are easy to score for a classifier are the clear-cut cases. Paraphrase, for example, can often be classified as correct and off-topic answers as incorrect. At the same time, entailment classes that contained both correct and incorrect answers were also problematic for the classifier, i.e., the classifier struggles for cases that can also be unclear for humans.

TEXT:
Sent$_i$: The Hessian government wants to prevent this reform,
because when it comes to apple wine all Hessians agree.
Sent$_{i+1}$: Its easier for other apple wine nations like France or Spain.
Sent$_{i+2}$: There, the beverage is called Cidre or Sidra and may keep that name, because the term wine is not part of the name.
(...)

QUESTION:
o other European countries experience similar problems as Hesse. Why?

TARGET ANSWER:

- No, [**in other apple wine nations like France or Spain the beverage is called Cidre**] or Sidra and may keep that name, because the term wine is not part of the name.

LEARNER ANSWERS:

- [correct] No. other countries, like [**France or Spain, have other name for apple drinking, like Cidre.**]
- [incorrect] Against - the Hessian government should this reform.

Figure 5.8: Example of a reading text, connected question, and learner and target answers from CREG. The extracted paraphrase fragments between the target answer and the correct learner answer are in bold-print and square brackets.

5.4 Experimental Study 3: Using Statistical Alignments for Automatic Scoring

Studies 1 & 2 in this chapter extended a baseline using knowledge-based alignments between learner and target answers following Meurers et al. (2011c) with text- and entailment-based features. The restriction to sentence-sized units in that study is one limitation addressed by our current study. To overcome this problem, the following study presents an ASAS approach that uses semantic information based on *statistical alignments* to extract paraphrase fragments between learner answers and target answers which in turn are used to extract features for ASAS.

Central to this approach is a method that provides information about paraphrase relations between (parts of) student answer and target answer. It is based on the assumption that correct learner answers and target answers are ideally (but not always, as we have shown in our corpus study on the entailment relations between learner and target answer in Section 4.1) paraphrases of each other. If this is not the case and the learner answer entails the target answer or vice versa, there are often still parts of the learner and target answer that paraphrase each other. Consider

the example in Figure 5.8: Although target answer and correct learner answer are clearly not paraphrases of each other – the learner answer is less specific than the target answer, i.e., we would have a case of *reverse entailment* in the terminology of Corpus Study 1 – parts of the target answer are a paraphrase of parts of the learner answer (in bold-print in the example).

For extracting paraphrases, we adopt the approach of Wang and Callison-Burch (2011) and Regneri and Wang (2012), who extract sub-sentential paraphrase candidates ("paraphrase fragments") from monolingual parallel corpora, making essential use of GIZA++, a word alignment algorithm originally developed for aligning bilingual parallel texts in Machine Translation (Och and Ney, 2003). The alignment algorithm learns semantic information from the corpus in an unsupervised way, without any labeled training material. Once this semantic information is given, paraphrase fragments are predicted in a robust manner, using no or (in the chunk-based version of the algorithm) only very shallow additional linguistic information.

We create a parallel corpus from the CREG corpus in a rather straightforward way by providing sentence pairs that consist of, e.g., a learner answer and the corresponding target answer. We train a paraphrase fragment recognition system on this corpus following the approach by (Wang and Callison-Burch, 2011). The detected paraphrases are then used to assess the correctness of the learner answers in the CREG corpus. We do so by extracting features from the paraphrase fragments detected between a learner answer and the target answer and use these features as input to a linear regression learner. We consider features that are indicators for the strength of the semantic connection. The rationale is that a learner answer that shares no paraphrase fragment with the target answer is likely to be false, whereas a learner answer – target answer pair whose fragments are strongly linked is likely to involve a correct learner answer.

Goals of this Study

We address with this study the general research question RQ 2.1:

RQ 2.1: Can statistical alignment for prompt-independent scoring be used as an alternative to knowledge-based alignment?

In more detail, we investigate the following questions:

- Can we apply statistical alignment methods for paraphrase fragment detection to identify partial paraphrases between learner answers and their corresponding target answers?
- How is the scoring performance, if we use properties of those paraphrase fragments as features in an ASAS system?

Contributions

To our knowledge, we are the first to use automatic paraphrase fragment detection (and associated methods from machine translation) for the short answer scoring task. This method enables access to semantic knowledge in a robust and (almost) unsupervised way which is transferrable to other languages or domains with minimal additional effort. Evaluation on the CREG Corpus shows that information provided by paraphrase detection alone leads to quite good scoring results. More importantly, combining the system with shallow and deep semantic state-of-the-art systems leads to consistent performance gains. A combination of all three systems results in an accuracy of 88.9%, which surpasses the state of the art and seems to be appropriate for practical application.

5.4.1 Related Work

Related work on ASAS in general is presented in Section 2.4. This section focuses on previous work directly related to this study and relevant related approaches from the field of paraphrase and paraphrase fragment extraction.

As in our previous studies, we compare our work to our reimplementation of the alignment-based approach by Meurers et al. (2011c) (see Section 5.2.1). They report an accuracy of 84.6% on the CREG corpus. Our reimplementation reaches an accuracy of 86.8% for 10-fold cross-validation using a linear regression classifier.

We will also compare to the only deep semantic approach to short-answer scoring known to us described in Hahn and Meurers (2012). They provide an interesting solution to the robustness problem: as a semantic formalism they use Lexical Resource Semantics (LRS), which is a formalism enabling arbitrary degrees of underspecification, and a syntax-semantic interface using atomic dependency information. In effect, this guarantees that some kind of semantic representation is computed for any (grammatical or ungrammatical) input expression. The LRS representations for target and learner answer are aligned, and alignment features are extracted and used by a classifier. They reach state-of-the-art accuracy of 86.3% on the CREG corpus, with a system that requires hand-coded language-specific semantic knowledge.

A widely used method for paraphrase detection is the extraction of equivalent sentences from either parallel or comparable monolingual corpora (Barzilay and McKeown, 2001; Barzilay and Elhadad, 2003; Quirk et al., 2004). However, for many NLP applications, sentences may turn out to be an impractical unit for paraphrasing, as the situation that two sentences convey exactly the same meaning is rather rare.

Recently, the research focus for paraphrase extraction has therefore been expanded to also consider sub-sentential paraphrase fragments as units of analysis that are not restricted to a

particular category. This is done to account for partial semantic overlap between sentences that can be expressed using various types of categories, as, e.g., *her preference* vs. *what she prefers*.

Recent approaches to paraphrase fragment extraction include Bannard and Callison-Burch (2005), Zhao et al. (2008) and Wang and Callison-Burch (2011). As pure word matching is not enough to achieve good results, most systems include syntactic information in the form of constituent or dependency structures (Callison-Burch, 2008; Regneri and Wang, 2012).

Gleize and Grau (2013) apply sentential paraphrase identification for scoring student answers. Their method is based on substitution by Basic English variants. They project the actual form of the answers onto a simple language and argue that in this way it is easier to draw inferences. However, by the mapping to the simplified representation not the entire semantic content is transferred. In addition, this method relies on available resources like dictionary and some hand-crafted rules, which is problematic when dealing with low resource languages.

5.4.2 Experiments and Results on Paraphrase Fragment Detection

This section describes our work on detecting paraphrase fragments in the context of reading comprehension exercises for learners of German as a foreign language. After describing the corpus data (Section 5.4.2) and the method (Section 5.4.2), we present an evaluation and analysis of the paraphrase fragments we detect (Section 5.4.2 and Section 5.4.2).

Data

As in our previous studies in this chapter, we use the balanced subset of the Corpus of Reading Comprehension Exercises in German (CREG) (Ott et al., 2012), first for paraphrase fragment detection and later (see Section 5.4.3) as a testbed for using the extracted fragments in a short answer grading scenario.

In Corpus Study 2 in Section 4.2, we extended CREG with a set of annotations linking each target and learner answer to its likely source sentence in the text. We make use of these annotated gold sentences when building a comparable corpus.

Method

Wang and Callison-Burch (2011) and Regneri and Wang (2012) (abbreviated as WCB&RW in the following) describe a procedure for extracting paraphrase fragments which consists of the following steps: constructing a parallel/comparable corpus, estimating word alignments over this corpus, computing positive and negative lexical associations, refining the alignment and, finally, detecting paraphrases. We follow this general method, customizing some steps to suit the needs of our application context.

For paraphrase fragment detection, we present two versions of our system: *basic*, which uses only word alignments for the detection step, and *chunk-based*, which also makes use of shallow syntactic analysis.

Building a Comparable Corpus. The aim in building a comparable corpus is to collect pairs of sentences which are likely to contain paraphrase fragments. To build our collection of sentence pairs, we exploit properties of the short answer grading scenario (via the CREG corpus).

Target answers (TA) and correct learner answers (LA) are the first, most obvious candidate pairs, as they convey the same meaning. We also include TAs paired with incorrect LAs. Such pairs are sometimes completely unrelated, thus introducing noise to the data, but sometimes they overlap enough to share one or more paraphrase fragments. Our aim is specifically pairs of sentences. In cases where an answer consists of more than one sentence, we include all possible combinations of TA sentence and LA sentence. This expands the number of sentence pairs, but also introduces additional noise.

In order to provide a richer source of lexical variation, we extend the input with pairs consisting of a TA or LA and its corresponding sentence from the reading text: In Section 4.2, we describe both human annotations of the best fitting sentence from the reading text for an answer and a procedure for automatically identifying the most closely-linked text sentence. We use both also in the experiments described below: the *goldlink condition* uses human annotations, and the *autolink condition* takes the sentence which has the highest alignment weight to the answer when the two sentences are aligned using the method described in (Meurers et al., 2011c).

We thus arrive at an input corpus, consisting of five sub-corpora: TA – *correct* LA, TA – *incorrect* LA, TA – *text sentence*, *correct* LA – *text sentence*, and *incorrect* LA – *text sentence*.

We increase the training material available by boosting the corpus in several ways. First, to emphasize the importance of lexical identity for learning word alignments, we add trivially-identical pairs: each reading text sentence paired with itself, and each word in the CREG corpus vocabulary, also paired with itself. Additionally, we repeat non-identical sentence pairs, with the number of repetitions linked to the nature of the sub-corpus in which the pair appears. We have also begun experiments adding word pairs from GermaNet (Hamp and Feldweg, 1997), in order to learn lexical paraphrases, but the results reported here do not include GermaNet-based boosting.

For intrinsic evaluation of the detected paraphrase fragments (Section 5.4.2), we aim to reduce noise in the data and emphasize reliable sentence pairs. Accordingly, each pair involving correct LAs, as well as those with TAs and text sentences, is copied 10 times. Pairs involving incorrect LAs appear just one time. The trivially-identical pairs are entered 10 times for sentences and 20

times for word pairs.

Preprocessing. To prepare the data for word alignment, we apply a standard linguistic preprocessing toolchain, consisting of sentence segmentation using `OpenNLP`,[5] tokenization with the `Stanford Tokenizer`,[6] lemmatization and part-of-speech (POS) tagging, both using the `TreeTagger` (Schmid, 1995). We use the `Stanford Named Entity Recognizer`[7] to identify persons, organizations, locations and dates. For robustness against grammatical errors and to reduce vocabulary size, all tokens are replaced with their lemmatized forms. We replace all occurrences of NEs with the corresponding NE-tag (e.g., *PERSON*). We treat spelling errors as described for the baseline model in Section 5.2.1.

Detecting paraphrase fragments. Following WCB&RW, we pass our input corpus to GIZA++ (Och and Ney, 2003) in order to: (a) estimate word alignments for input sentence pairs, and (b) obtain a lexical correspondence table with scores for individual word pairs.

Links between aligned words in the sentence pairs are then classified as positive or negative based on their scores, a technique which has previously been applied to extract paraphrase fragments from non-parallel bilingual corpora and has been shown to improve a state-of-the-art machine-translation system (Munteanu and Marcu, 2006). Word pairs containing punctuation or stop words are excluded from the alignment prior to scoring.[8]

Afterwards, the alignment is refined by removing all negatively-scored word pairs, such that only very strong alignments survive. We then smooth the alignment by recomputing scores for each word, averaging over a window of five words. In this way we often capture context words that are left out of the alignment process (e.g. determiners, prepositions, or particles) but are nonetheless necessary for producing linguistically well-formed fragments.

For the *basic* version, a source-side fragment is detected by extracting sequences of adjacent words with positive scores after smoothing. The corresponding target-side fragment is induced using one of two methods: The *unidirectional* approach finds the target fragment by using the lexical scores for the source side plus alignment links to the target side. In the *bidirectional* approach, we also compute lexical scores for the target side and extract target-side fragments in that manner.

Despite the use of smoothing for producing more grammatical fragments, the basic approach often produces output of questionable readability, e.g., ”hm, so” is a fragment that lacks context in order to understand the intended semantic content. Especially if these fragments might be used

[5] http://opennlp.apache.org/
[6] http://nlp.stanford.edu/software/tokenizer.shtml
[7] http://nlp.stanford.edu/software/CRF-NER.shtml
[8] http://www.ranks.nl/stopwords/german.html

to give feedback to learners, it is important to produce readable output. This is the motivation for the second version of the system.

In the *chunk-based* version we reset the boundaries of the basic fragments in a post-processing step by taking syntactic chunk information into consideration. If a fragment has some overlap with a chunk, then the remainder of that chunk is also included in the fragment. We also apply some heuristics to account for aspects of the German language: e.g., prefixes of separable verbs and past participles often appear in sentence-final position and should be covered by the fragment.

The fragment extracted from the source sentence is the same for all configurations but the target fragment differs. Example 5.2 illustrates the difference of fragments extracted by the unidirectional vs. bidirectional method and example 5.3 the one of basic vs. chunk-based.

(5.2) **source fragment**: in front of the PC or the TV
target fragments:
`uni`: with the PC or the TV
`bi`: all time with the the PC or the TV

(5.3) **source fragment**: in vegetable garden one has to chop and water
target fragments:
`basic`: in vegetable garden chop and waterz
`chunk`: one can chop and water in vegetable garden

An interesting observation is that the bidirectional method tends to be too greedy. Target fragments returned with it contain additional information that has no corresponding part on the source side. The chunk-based system is useful because it augments a fragment but also slightly modifies its semantic content.

Intrinsic Evaluation of Detected Paraphrases

To evaluate precision of the extracted paraphrases, we again follow WCB&RW. For each of the two systems, 300 fragment pairs are randomly extracted, half with the unidirectional version and half with the bidirectional. These are evenly distributed across LA-TA pairs and answer-text sentence pairs. Each fragment is labeled by two annotators with one of four categories: paraphrase, related, unrelated, or invalid. The label *related* is assigned when there is overlap between the two fragments, but they are not paraphrases, and *invalid* is assigned if one or both fragments are completely ungrammatical or not readable. Annotators were not told the type of the sentence pair, and they were instructed to ignore spelling and grammatical errors in evaluating paraphrases.

	4 categories	**2 categories**
basic	0.22 (fair)	0.69 (good)
chunk-based	0.52 (moderate)	0.84 (very good)

Table 5.9: Inter-annotator-agreement

	unidirectional	**bidirectional**
basic	0.78	0.74
chunk-based	0.69	0.71

Table 5.10: Precision of paraphrase fragment detection

Table 5.9 shows the inter-annotator agreement in two conditions: if we consider all four labels separately, and if we instead merge *paraphrase* and *related* as well as *unrelated* and *invalid*. Results are along the lines of (Regneri and Wang, 2012) who report kappa values of 0.55 for four-label annotation and 0.71 for a two-label condition. Our basic system shows worse agreement than the chunk-based. This is due to the fact that basic fragments are often linguistically not well-formed and are therefore harder to annotate. For the final gold standard, all conflicts have been resolved by a third annotator. This gold-standard annotation is then used for evaluating the quality of the fragments. For measuring the precision of the extracted paraphrases, i.e., for measuring what percentage of the fragment pairs identified should indeed be considered as paraphrases or related, we use the two-label condition. Results are presented in Table 5.10. Precision on our data set is in the same range as that reported by Wang and Callison-Burch (2011) (62 to 67%) on a monolingual comparable corpus. Overall the performance of the basic system is better than the chunk-based. This is an unexpected result because the chunk-based system was developed specifically to improve the quality of the basic fragments. However, missing tokens like prepositions that are added to a fragment by the chunk system can change its meaning and as a consequence the fragments are no longer related. Between the unidirectional and bidirectional approaches there is no statistically significant difference, according to a chi-squared test (Pearson, 1900).

For the application of the extracted paraphrase fragments to short answer scoring, the unidirectional approach is used, because it gave us the best results for the generally better *basic* version of the system.

We expect variability across correct and incorrect answers, because in scoring a learner an-

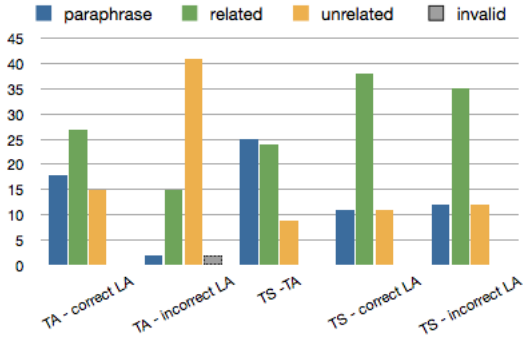

Figure 5.11: Distribution of annotation labels for the five subcorpora. TA stands for target answer, LA for learner answer and TS for the corresponding text sentence.

swer, strict paraphrases are not always necessary. For example a question in the corpus asking *"Wer war an der Tür"* (*"Who was at the door?"*) with the target answer *"Drei Soldaten (three soldiers) waren an der Tür"* the learner answer *"Drei Männer (three men) waren an der Tür"*, although less specific, was also graded as correct by the teachers. To investigate this variability, we look at the distribution of the four categories across the various subcorpora.

Figure 5.11 depicts the distribution of the labels – exemplarily for the chunk-based version – showing how often each annotation label occurred within the five subcorpora TA – *correct* LA, TA – *incorrect* LA, TA – *text sentence*, *correct* LA – *text sentence*, *incorrect* LA – *text sentence*. We can see that correct learner answers lead to substantially more paraphrases of the target answer (18) than do incorrect learner answers (2). Incorrect learner answers, however, have a much higher degree of unrelated fragments with the target answer (41 vs 15). Invalid fragments are in general very rare.

In the subcorpora involving text sentences, both correct and incorrect learner answers have a similarly high degree of paraphrase and related cases. That is the case because both correct and incorrect learner answers are often paraphrases of some part of the text. In the case of a correct answer, the target answer is often a paraphrase of the same text sentences as the text sentence for the learner answer. In the case of an incorrect learner answer, the student often erroneously paraphrased a text sentence that has nothing to do with the correct answer (as we have seen in Section 4.2).

sub-corpus	productivity in %
ta-corr la	95
ta-incorr la	**78**
textSent-ta	95
textSent-corr la	94
textSent-incorr la	92
total	91

Table 5.12: Productivity by subcorpus

Analysis of the Detected Fragments

This section presents continued analysis of the detected fragments from various subcorpora, covering productivity and variability of lexical material.

Productivity of the Detected Fragments Table 5.12 shows productivity by subcorpus, measured by how often at least one fragment pair is detected per input sentence pair. As expected, productivity is lowest for incorrect LAs paired with TAs. Incorrect LAs paired with text sentences, however, show productivity similar to other subcorpora. This is not surprising, as incorrect learner answers often stem from some part of the text, as we have seen in Corpus Study 2, although not necessarily the same as the target answer.

Lexical Variety of the Detected Paraphrases In a next step, we evaluate whether our paraphrase fragments are able to cover different lexicalizations of the same content. Inspection of the data shows that our approach indeed detects real paraphrase fragments, beyond the trivial case of identical spans of text in paired sentences.

To quantify lexical variety, we measure the degree of lemma overlap between sentence pairs and fragment pairs extracted from these sentence pairs. Figure 5.14 shows that there is a significantly higher overlap between paraphrase pairs than between sentences. This is an expected result, but on the other hand, the overlap is not so extensive that it makes the paraphrase detection task trivial.

Table 5.13 shows example fragments detected by the chunk-based, unidirectional method. The qualitative analysis shows that non-identical material contained in the fragments often captures alternative expressions of the same semantic content. However, we can see that the method would benefit from handling of phenomena such as negation, antonymy, or relatedness between

Example	**source**	**target**
1	Die Stadtverwaltung sagt nein	Die Stadtverwaltung ist dagegen
2	kein glücklisches Ende	ein schlechte Ende
3	die Broadway-Version erhielt sechs Tonys	Es hat sechs Tonys gewonnen
4	Damit lachen die anderen Kinder sie ja aus	die anderen Kinder lachen Julchen aus
5	darf nicht mehr verwendet werden	dann nicht mehr erlaubt
6	Die Leute wissen *nicht*	die meisten Leute wissen
	ihre genauen monatlichen Ausgaben	wie eine Budgetplan zu machen
7	in einem *Neu*bau	in einem *Alt*bau
8	würde mit *Computer* arbeiten	würde mit *Wissenschaftlerin* arbeiten
9	[Nicht, sagten die Augen] der Frau, nicht lachen	[Er sollte nicht] lachen, weil das Kind [schlief]

Table 5.13: Exemplary Fragments output with the *unidirectional* method for the *chunk-based* system

nouns or other content words.

Fragment pair 7 illustrates the difficulty faced in cases where antonymy is present. The compound words "Altbau" and "Neubau" both carry the main meaning of a building (*der Bau*) and are therefore related, but the modifying words "alt" and "neu" (*old* and *new*) are antonyms. They are identified as paraphrase fragments, because both terms occur in very similar contexts, i.e., because they are used to answer the same question.

Fragment pair 8 again highlights this problem when word alignments like *"Computer-Wissenschaftlerin (scientist)"* are learned, even though they are not valid paraphrases and the word *"Wissenschaftlerin"* only occurs in incorrect answers. This happens in cases when an input sentence pair shares many identical words, and one or more non-identical words that occur very infrequently (or even nowhere else) in the corpus. In such a case, GIZA++ learns strong alignments between the identical words and also between the two unrelated words, as there are no other options for linking those words.

The last fragment pair 9 shows an example of unrelated fragments, which are probably (mistakenly) classified as paraphrases because of the high token overlap.

5.4.3 Experiments and Results on Short Answer Scoring

We use the results of the paraphrase fragment detection (section 5.4.2) as the basis for automatic short answer scoring. In this section we describe our method and evaluate it on the CREG corpus.

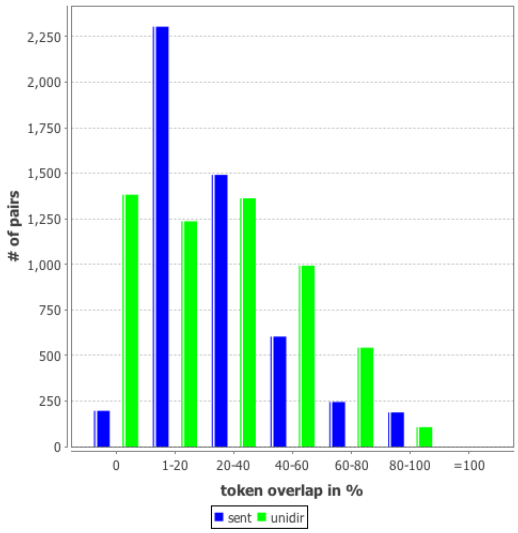

Figure 5.14: Percentage of identical tokens in sentence pairs (sent) and fragment pairs (unidir)

Method

We base the assessment of the correctness of the learner answer on the paraphrase relation between learner answer and target answer. We take the case when no paraphrase is found to be strong evidence against correctness. If a paraphrase pair is detected, we want to make the scoring decision dependent on properties of the single paraphrases and their interrelation. Technically, we use a binary classifier, which bases its prediction on features extracted from the paraphrase fragments. Concretely, we employ the linear regression classifier from the Weka Toolkit (Hall et al., 2009).

As outlined in the introduction, we employ different modes to identify pairs of LA and TA paraphrases: In the direct mode, we directly determine a TA-LA paraphrase pair based on the alignment between LA and TA. In the indirect mode, we pair each of LA and TA with a text sentence (these may be identical or different sentences), independently derive paraphrase pairs for LA with text sentence and TA with text sentence, respectively, which in the success case gives us a TA-LA paraphrase pair obtained in an indirect way. We assume that the indirect mode provides additional information through the relatedness between LA, TA, and the text.

For each of the two comparison modes a set of features is extracted, which provide information about the relation between the paraphrase fragments f_1 and f_2, which are extracted from a sentence pair s_1 and s_2 or about single fragments.

The following features are considered:

1. token overlap: jaccard coefficient $J(tokens(f_1), tokens(f_2)) = \frac{|tokens(f_1) \cap tokens(f_2)|}{|tokens(f_1) \cup tokens(f_2)|}$, 0 if there are no fragments
2. difference in fragment length $|f_1 - f_2|$, -1 if there are no fragments
3. percentage of tokens in the s_1 covered by f_1
4. percentage of tokens in the s_2 covered by the f_2
5. average of lexical scores for the target answer (resulting from word alignment)

Because we use the unidirectional alignment version and take the text sentence to be the source sentence, only lexical scores for the text sentences are computed in the indirect case. Therefore the fifth feature is not available in the indirect mode.

Evaluation

We compare our approach to both the alignment model (as in (Meurers et al., 2011c) and also used in previous studies in this chapter) and the deep semantic model by (Hahn and Meurers, 2012). We re-implement the alignment model using features for token and chunk alignment reaching an accuracy of 86.8% on the CREG corpus (compared to 84.6%in the (Meurers et al., 2011c) model). The deep semantic model reaches an accuracy of 86.3%, also on the CREG data. We make direct comparison against these two scores; a random baseline for this balanced data set is 50%.

We evaluate using tenfold cross-validation, running the complete paraphrase fragment detection method (Section 5.4.2) on nine folds for training. For the test corpus, of course, we do not know ahead of time whether answers are correct or not. Thus we build our input corpus without taking advantage of this information. In this setting, each pair involving a LA or TA is included 10 times, regardless of the correctness of the answer.

We evaluate our model alone and using additional features from the other two models, as is shown in Table 5.15: In order to see the contribution of the direct and indirect feature sets, we evaluate those sets individually (*paraphrases direct* and *paraphrases indirect*) and together (*paraphrases combined*). For combining with the other models, we always use the combined set of paraphrase features.

To evaluate our model in combination with the alignment model (*paraphrases + alignment system*), we add the features from our reimplementation. We also combine our model with both of the other two models (*paraphrases + alignment model + deep semantics*), using the semantic scores obtained by Hahn and Meurers (2012) as an additional feature.

Evaluation Corpus	paraphrases direct	paraphrases indirect	paraphrases combined	paraphrases + alignment	paraphrases + deepSemScore	paraphrases + deepSemScore + Alignment	alignment + deepSemScore
autolink - basic	76.9	70.6	78.3	86.5	86.9	87.7	87.5
autolink - chunk	76.8	70.1	77.1	86.4	86.7	88.1	87.5
goldlink - basic	77.5	72.8	77.6	86.5	87.0	88.1	87.5
goldlink - chunk	76.6	72.1	77.4	86.7	87.1	**88.9**	87.5

Table 5.15: Accuracy on CREG balanced corpus with various model combinations

Table 5.15 summarizes our results: We can see that our system alone, while being far from reaching the state of the art, can reasonably differentiate between correct and incorrect answers. The direct comparison of learner answer and target answer (*paraphrases direct*) works better than just the indirect comparison via fragments obtained from alignment with the text. In combination, the indirect features still contribute to the performance *paraphrases combined*, although not in a statistically significant way.

When combining the paraphrase features with the features from the alignment system, we don't get an improvement over the alignment system (86.8%). When additionally adding the semantic score to both feature sets, we reach our best result with an accuracy of 88.9 % which is not significantly better (α=0.25) according to a McNemar test) than the comparison figure of 86.8%.

When comparing the *goldlink* to the *autolink* condition, we see an advantage of having the optimal information about the best matching sentence in the indirect feature set.

There is no clear trend as to whether the *basic* or the *chunk-based* system performs better. The paraphrase fragments model on its own is not good enough to beat the other methods. However, combining the three systems gives an improvement of 2.1%, which is an indication of complementary information provided by the different feature sets.

5.4.4 Conclusions

In this study, we have presented the first approach which uses paraphrase information for automatically scoring short answers. We successfully adapt a paraphrase fragment extraction method to the new domain of reading comprehension data for learning German as a foreign language. In this way, we frame the short answer scoring task with respect to semantic information that is robust to noise in the input. Because of this robustness, and because of its (nearly) unsupervised nature, the approach is readily adaptable for other languages or domains. We obtain good scoring results using detected paraphrases, and when we combine our method with shallow and deep semantic systems, we surpass the state of the art on the CREG corpus.

We see four obvious extensions for future research. First, paraphrase fragments detected between target and learner answers, or between learner answers and the reading text, could be very useful in practical educational applications, such as providing direct feedback to language learners. This could be done by highlighting for a learner the paraphrased regions of his answer and, more importantly, those which do not stand in such a semantic relationship to the target answer or the text. Second, we are interested in investigating the influence of information structure on scoring; fragments which cover information from the question should receive less weight than fragments which offer new information, and our fragment detection method is one way of making such distinctions. Third, our method can be adapted to handle on-line input, computing alignments based on previously-existing lexical correspondence tables and in this way providing immediate output for new learner answers. Finally, when creating the parallel corpus as the basis for learning statistical alignments, we currently exploit the assumed paraphrase relation between learner answers and their corresponding target answers. However, also all correct learner answers to the same question should be at least partially paraphrases of each other. Therefore one could also extend the parallel corpus by such pairs of correct learner answers in order to create larger training data sets.

5.5 Conclusions

This section presented three studies on automatic short-answer scoring. They use findings from the previous corpus studies and helped us to gain insight on the nature of short answer questions: In Study 1, we found it confirmed that the link between learner answers and the reading text is so strong that it can be used in a simple baseline classifier. However, features based on the relation to the text improved an alignment-based classification model only marginally, showing that the link to the text and the link to the target answer – which itself links to the text –cover similar information.

We observed in Study 2 that textual entailment and ASAS are indeed not only theoretically related, but that one feature set can be used to learn both tasks. Finally, we have seen in Study 3 that paraphrase relations between learner and target answer occur on the sub-sentential level. With statistical alignment, we have described a way how to compute an ASAS model by using methods from statistical machine translation. Although this model uses a very different kind of alignment than the knowledge based model, it reaches a comparable performance.

These findings might influence practical applications of ASAS in future work. A scorer exploiting properties of the text could be used in different ways. First, students could be asked to mark relevant sentences in a text (e.g., in an on-line learning platform) and receive feedback upon that prior to formulating their answer. Such an approach would be an easier task that could

help weaker students if used as precursor to the main task; at the same time it could support automatic scoring, as it would rule out those incorrect answers where a student extracted information for her answer in the wrong place. Alternatively, student might also request from the system a hint as to where to look for an answer in the text.

Information on paraphrase fragments might also be used in a real life system to give informative feedback to students by informing them about which passages can be aligned to a target answer and which not.

On the more theoretical side, eye-tracking studies could be used to determine where in the text a student looked for an answer and would shed more light as to what sentences might have been inspected as a source for the answer, even if the actual answer the student wrote comes from the right portion of the text.

In the next chapter, we continue with more application-related experimental studies, where we have the perspective of a teacher in mind whose workload for correcting short-answer questions will be reduced in different ways.

6 Experimental Studies – Computer-Assisted Scoring

Fully automatic short-answer scoring (as addressed in the previous chapter) aims at reducing human scoring effort by training a classifier that is then used to automatically label all answers in a data set. But this is only one way of reducing a teacher's workload. In this chapter we present studies addressing a slightly different approach - leveraging semi- and unsupervised machine learning methods to directly interact with teacher scoring. Our aim is to reduce human scoring effort, therefore we investigate how to optimize scoring performance such that only a limited number of scoring steps is necessary.

This chapter presents the second group of experimental studies in this thesis. After studies on automatic short-answer scoring in Chapter 5, the experiments in Chapter 6 focus on semi-automatic scoring of short-answer questions with the goal of reducing human annotation effort. Most existing automatic SAS systems rely on supervised machine learning techniques, where a classifier is learned from labeled training data. Thus, instead of manually labeling all student answers in a data set, a teacher would label only a subset and train a classifier on this set for labeling the complete data set. As for most supervised learning scenarios, automatic SAS systems perform more accurate scoring as the amount of data available for learning increases. Particularly in the educational context, though, simply labeling more data is an unsatisfying and often impractical recommendation. New prompts have to be generated frequently to avoid that learners might know test questions from previous years, and there is a need for automatic scoring approaches that can do accurate assessment with much smaller amounts of labeled data.

We investigate methods for efficient scoring without relying on previously-trained classifiers that are applied to new data. One obvious way to reduce scoring effort for a teacher is to re-use a classifier trained for a specific question by re-using the same question in a new classroom setting and simply applying the existing classifier to the new unseen test data for the same prompt. Such an approach is impractical for the above mentioned reason that students might know questions in advance. Previous work addressed the question of human scoring effort reduction by investigating how a prompt-independent classifier, like the ones considered in Chapter 5, can be applied to new sets of questions. Such an application scenario was considered in the SemEval

Student Response Analysis Task (Dzikovska et al., 2013) in two of the three evaluation data sets, one containing questions not seen during training, but from the same educational domain, and one considering questions from a different domain. The results contradict intuitions and out-of-domain questions were sometimes graded more accurately than answers to questions in the training data, possible due to particular properties in the data. We do not consider such an approach here.

Instead, we have an application scenario in mind, where a batch of learner answers to a new prompt has to be scored and the question is, how this can be efficiently done by a teacher, i.e., how annotation effort is most efficiently allocated. Our work contributes to recent work investigating the influence of the quantity and quality of training data for ASAS (Zesch et al., 2015; Heilman and Madnani, 2015; Basu et al., 2013, among others).

We address this problem in three studies:

- In our Active Learning Study in Section 6.1, we investigate the applicability of Active Learning (AL) methods on the ASAS task. The core idea in AL is that training items are selected from a larger pool of training instances in such a way that a classifier learned from these instances has a better performance than a classifier learnt from the same number of randomly drawn training instances. We investigate different variants of AL on the ASAP data set and find that uncertainty-based sample selection methods perform best.

- In our Clustering Study 1 in Section 6.2, we present experiments on reducing human scoring effort by clustering similar answers together. The core idea in clustering is that similar answers are likely to be either both correct or incorrect. Therefore a teacher can score a cluster of answers by assigning just one label to all of them. We use label-propagation, where the teacher labels just one member of a cluster, which is then propagated to all other members of the cluster. We propose and evaluate several label-propagation methods and find that we can substantially reduce human scoring effort if we accept small numbers of mislabeled items.

- In Clustering Study 2 in Section 6.3, we add supervision to the traditionally unsupervised clustering process in order to make better use of the human annotations. While Clustering Study 1 used manually assigned labels only for the labeling of certain items within a cluster, we use these labels here also to tune the distance metric used in the clustering process and as constraints in semi-supervised clustering. This method improves scoring performance compared to unsupervised clustering, while not reaching the performance of supervised machine learning.

An added value of clustering is that it provides valuable structural information, while machine learning classifiers just assign a score. As an example, automatic clustering of the answers for the question *"What is one right or freedom from the First Amendment of the U.S. Constitution?"* from the PG data set yields different groups of correct and incorrect answers, respectively, such as *"{freedom of speech, the right of free speech, to have freedom of speech, ...}"* or *"{freedom of religion, freedom to practise religion, the freedom of religion, ...}"*, *"{to bear arms, the right to bare arms, right to arms, ...}"*. The first two clusters contain correct answers referring to different facts, the last one contains answers making the same error. Teachers may use the output clusters to identify common misconceptions among students and assign feedback to whole groups of answers.

6.1 Active Learning for Short-Answer Scoring

This study presents a way of reducing human annotation effort by selecting training instances for manual annotation in a way such that a classifier profits most from them compared to randomly selected training instances.

Zesch et al. (2015) present an approach investigating whether carefully selected training data in an SAS task lead to a better classification performance. For each prompt, they first cluster the entire set of responses and then train a classifier on the labeled instances that are closest to the centroids of the clusters produced. The intuition – that a training data set constructed in this way captures the lexical diversity of the responses – is supported by results on a data set with shorter responses, but on the ASAP data set, the approach fails to improve over random selection.

The natural next step – and the approach we pursue in our experiments on the ASAP data – is to use active learning (AL, Settles (2012)) for informed selection of training instances. In AL, training corpora are built up incrementally by a successive selection of instances according to the current state of the classifier (for a detailed description see Section 6.1.3). In other words, the machine learner is queried to determine regions of uncertainty, instances in that region are sampled and labeled, these are added to the training data, the classifier is retrained, and the cycle repeats. Training data sets sampled in this way typically perform better than a baseline of randomly selected instances, as highlighted in Figure 6.1.

Our approach differs from that of Zesch et al. (2015) in two important ways. First, rather than selecting instances according to the lexical diversity of the training data, we select them according to the output of the classifier. Second, we select instances and retrain the classifier in an incremental, cyclic fashion, such that each new labeled instance contributes to the knowledge state which leads to selection of the next instance.

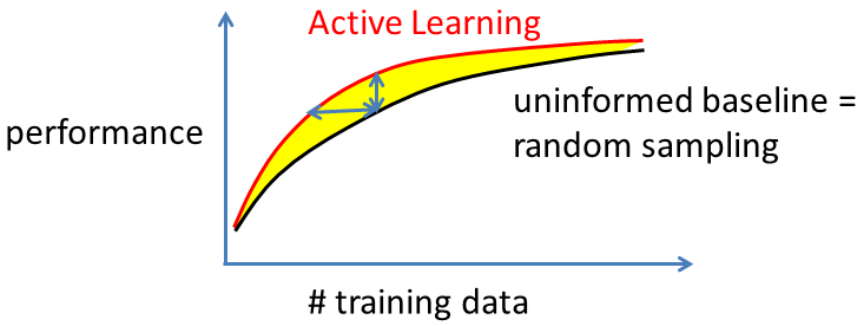

Figure 6.1: The core idea of AL: An AL method achieves better performance than a random baseline, highlighted by the yellow area between the AL and the random baseline curve.

Goals of the Study

This study addresses research question 2.2:

RQ 2.2: Are active learning methods suitable in a prompt-specific ASAS scenario?

In particular we systematically investigate a wide range of AL settings. Sample selection via AL involves setting a number of parameters, and there is no single best-for-all-tasks AL setting. Thus we explore a wide range of AL scenarios, implementing a number of established methods for selecting candidates. We consider three families of methods. The first are uncertainty-based methods, which target items about which the classifier is least confident. Next, diversity-based methods aim to cover the feature space as broadly as possible; the cluster-centroid selection method described above is most similar to this type of sample selection. Finally, representativeness-based methods select items that are prototypical for the data set at hand.

To date, there are no clear guidelines for matching AL parameter settings to particular classification tasks or data sets. To better understand the varying performance of different sample selection methods, we present an initial investigation of two properties of the various data sets: class imbalance and language model perplexity.

Contributions

Our results show a clear win for uncertainty-based methods, with the caveat that performance varies greatly across prompts. Perhaps unsurprisingly, we see that uncertainty-based sampling brings stronger gains for data sets with skewed class distributions, as well as for those with more

cleanly separable classes according to language model perplexity.

In sum, active learning can be used to reduce the amount of training data required for automatic SAS on longer written responses without representative target answers, but the methods and parameters need to be chosen carefully. Further investigation is needed to formulate recommendations for matching AL settings to individual data sets.

6.1.1 Related Work

This study contributes to a recent line of work addressing the question of how to reduce workloads for human graders in educational contexts, in both supervised and unsupervised scoring settings.

One solution to this problem is to develop generic scoring models which do not require retraining in order to do assessment for a new data set (i.e., a new question/prompt plus responses). Meurers et al. (2011c) apply such a model for scoring short reading comprehension responses written by learners of German. This system crucially relies on features which directly compare learner responses to target answers provided as part of the data set, and the responses are mostly one sentence or phrase. In this work we are concerned with longer responses generated from a wide range of prompt types, from questions asking for list-like responses to those seeking coherent multi-sentence texts (details in Section 6.1.2). For such questions, there is generally no single best response, and thus the system cannot rely on comparisons to a single target answer per question. Rather systems need features which capture lexical properties of responses to the prompt at hand. In other words, a new scoring model is built for each individual prompt.

The work most closely related to ours is Zesch et al. (2015), which includes experiments with a form of sample selection based on the output of clustering methods. More precisely, the set of responses for a given prompt (using both the ASAP and Powergrading corpora) are clustered automatically, with the number of clusters set to the number of training instances desired. For each cluster, the item closest to its centroid is labeled and added to the training data. This approach aims at building a training set with high coverage of the lexical variation found in the data set. The motivation for this approach is that items with similar lexical material are expressed by similar features, often convey the same meaning and in such cases often deserve the same score. By training on lexically-diverse instances, the classifier should learn more than if trained on very similar instances. Of course, a potential danger is that one cluster may (and often does) contain lexically-similar instances that differ in small but important details, such as the presence or absence of negation.

For the ASAP corpus (which is also the focus of our experiments), the cluster-centroid sampling method by Zesch et al. (2015) shows no improvement over a classifier trained on randomly-sampled data. An interesting outcome of the experiments by Zesch et al. (2015) is the highly-

variable performance of classifiers trained on a fixed number of randomly-sampled instances; out of 1000 random trials, the difference between the best and worst runs is considerable. The performance variance of systems trained on randomly-selected data underscores the need for more informed ways of selecting training data.

In the domain of educational applications, AL has recently been used in two different settings where reduction of human annotation cost is desirable. Niraula and Rus (2015) use AL to judge the quality of automatically generated gap-filling questions, and Dronen et al. (2014) explore AL for essay scoring using sampling methods for linear regression.

To the best of our knowledge, AL has not previously been applied to automatic SAS. Our task is most closely related to studies such as Figueroa et al. (2012), where summaries of clinical texts are classified using AL, or Tong and Koller (2002) and McCallum and Nigam (1998), both of which label newspaper texts with topics. Unlike most other previous AL studies, text classification tasks need AL methods that are suitable for data that is represented by a large number of mostly lexical features.

6.1.2 Experimental Setup

This section describes the data set, features, and classifier used in our experiments.

Data

All experiments are performed on the ASAP corpus (see Section 3.3). We use answer sets for all 10 individual prompts. Although scores are numeric (0.0-2.0/3.0 in 1.0 steps), we treat each score as one class and model the problem as classification rather than regression. This approach is in line with previous related work as well as standard AL methods.

For each prompt, we split the data set randomly into 90% training and 10% test data. We then augment the test set with all items from the ASAP "public leaderboard" evaluation set. Table 6.2 shows the number of responses and label distributions for each prompt and each data subset. Some data sets (i.e., answer set per prompt) are clearly much more imbalanced than others.

Classifier and Features

In line with previous work on the ASAP data, classification is done using the Weka (Hall et al., 2009) implementation of a sequential minimal optimization algorithm (Platt, 1998) for training a support vector classifier.

For feature extraction, all answers are preprocessed using the OpenNLP sentence splitter[1] and the Stanford CoreNLP tokenizer and lemmatizer (Manning et al., 2014). As features, we use

[1] https://opennlp.apache.org/

	training					test				
prompt	**#answers**	**0.0**	**1.0**	**2.0**	**3.0**	**#answers**	**0.0**	**1.0**	**2.0**	**3.0**
1	1505	331	389	474	311	724	152	208	225	139
2	1150	150	289	422	289	554	86	137	190	141
3	1625	385	913	327	-	589	145	322	122	-
4	1492	571	803	118	-	460	190	232	38	-
5	1615	1259	291	37	28	778	594	138	27	19
6	1617	1369	143	60	45	779	644	73	41	21
7	1619	837	405	377	-	779	390	195	194	-
8	1619	501	418	700	-	779	224	204	351	-
9	1618	390	661	567	-	779	195	312	272	-
10	1476	261	688	527	-	710	110	348	252	-

Table 6.2: Data set sizes and label distributions for training and test splits. '-' indicates a score does not occur for that data set.

lemma 1- to 4-grams to capture lexical content of answers, as well as character 2- to 4-grams to account for spelling errors and morphological variation. We lowercase all textual material before extracting ngrams, and features are only included if they occur in at least two answers in the complete data set.

This is a very general feature set that: (a) has not been tuned to the specific task, and (b) is similar to the core feature set for most other ASAS work on the ASAP data. In preliminary classification experiments, we also tried out features based on skip n-grams, content-word-only n-grams, and dependency subtrees of various sizes. None of these features resulted in consistently better performance across all data sets, so they were rejected in favor of the simpler, smaller feature set.

6.1.3 Parameters of Active Learning

The core algorithm we use for active learning is the standard setting for pool-based sampling (Settles, 2010); pseudocode is shown in Figure 6.3. The process is vizualized in Figure 6.4.

The process begins with a pool of unlabeled training data and a small labeled seed set. At the start of each AL round, the algorithm selects one or more instances whose label(s) are then requested. In simulation studies, requesting the answer means revealing a pre-annotated label; in real life, a human oracle (i.e., a teacher) would provide the label. After newly-labeled data has been added to the training data, a new classifier is trained, run on the remaining unlabeled data, and the outcomes are stored. For uncertainty sampling methods, these are used to select the instances to be labeled in the next round. The classifier's performance is evaluated on a fixed

The AL algorithm

split data set into *training* and *test*
select seeds $s_0, s_1, ..., s_n \in$ *training*
request labels for $s_0, ...s_n$
labeled $:= \{s_0, s_1, ..., s_n\}$
unlabeled $:=$ *training*$\backslash\{s_0, s_1, ..., s_n\}$
while *unlabeled* $\neq \emptyset$:
  select instances $i_0, i_1, ..., i_m \in$ *unlabeled* $*$
  unlabeled $=$ *unlabeled*$\backslash\{i_0, i_1, ..., i_m\}$
  request labels for $i_0, i_1, ..., i_m$
  labeled $=$ *labeled*$\cup\{i_0, i_1, ..., i_m\}$
  build a classifier on *labeled*
  run classifier on *test* and report performance
$*$ according to some sample selection method

Figure 6.3: Pseudocode for general, pool-based active learning.

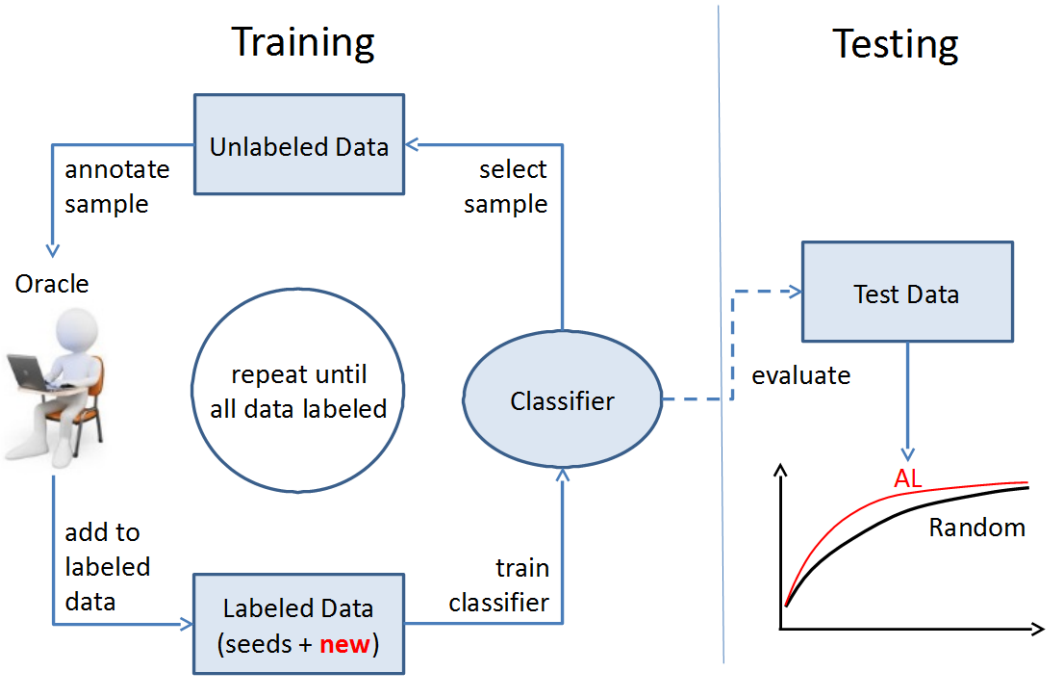

Figure 6.4: The active learning cycle following Settles (2012)

test set. The efficacy of the item selection method is evaluated by comparing the performance of this classifier to that of a classifier trained on the same number of randomly-selected training instances.

In the following, we discuss the main factors that play a role in active learning: the item selection methods that determine which item is labeled next, the number of seed instances for the initial classifier and how they are chosen, and the number of instances labeled per AL cycle.

Item Selection

The heart of the AL algorithm is (arguably) item selection. Item selection defines how the next instance(s) to be labeled are selected, with the goal of choosing instances that are maximally informative for the classifier. We explore a number of different item selection strategies, based on either the uncertainty of the classifier on certain items (*entropy*, *margin*, and *boosted entropy*), the lexical *diversity* of the selected items, or their *representativeness* with respect to the unlabeled data.

Random Baseline. We use a standard random sampling baseline. For each seed set, the random baseline results are averaged over 10 individual random runs, and evaluations then average over 10 seed sets, corresponding to 100 random runs.

Entropy Sampling is our core uncertainty-based selection method. Following Lewis and Gale (1994), we model the classifier's confidence regarding a particular instance using the predicted probability (for an item x) of the different labels y, as below.

$$x_{selected} = \arg\max_x \left(-\sum_i P(y_i|x) log P(y_i|x) \right)$$

Classifier confidence is computed for each item in the unlabeled data, and the one with the highest entropy (lowest confidence) is selected for labeling.

Boosted Entropy Sampling Especially for very skewed data sets, it is often favorable to aim at a good representation of the minority class(es) in the training data selected for AL. By a good representation, we mean one that selects the minority classes with a higher frequency than their occurrence in the training data. To do so, we adopt the method of boosted entropy sampling by Tomanek and Hahn (2009), where per-label weights are incorporated into the entropy computation, in order to favor items more likely to belong to a minority class. Tomanek and Hahn (2009) apply this technique to named entity recognition, where it is possible to estimate the true label distribution. In our case, since we don't know the expected true distribution of

scores, for each AL round, we instead adapt label weights using the distribution of the current labeled training data set.

Margin Sampling is a variant of entropy sampling with the difference that only the two most likely labels (instead of all three or four) are used in the comparison, i.e., we select an instance, where the difference in the probabilities between the most likely y_1 and second most likely class label y_2 is minimal:

$$x_{selected} = \arg\min_x \left(-P(y_1|x) - P(y_2|x)\right)$$

As a result, this methods tends to select instances that lie on the decision border between two classes, instead of items at the intersection of all classes.

Diversity Sampling-all aims to select instances that cover as much of the feature space as possible, i.e., that are as diverse as possible. We model this by selecting the item with the lowest average cosine similarity between the item's feature vector and those of the items x_i in the current labeled training data set.

$$x_{selected} = \arg\min_x \left(\sum_i \text{sim}(x, x_i)\right)$$

Representativeness Sampling uses a different intuition: this method selects items that are highly representative of the remainder of the unlabeled data pool. We model representativeness of an item by the average distance (again, measured as cosine similarity between feature vectors) between this item and all other items in the pool. This results in selection of items near the center of the pool.

$$x_{selected} = \arg\max_x \left(\sum_i \text{sim}(x, x_i)\right)$$

Note that these selection methods are somewhat complementary. While entropy and margin sampling generally select items from the decision boundaries, they tend to select both outliers and items from the center of the distribution.

Representativeness sampling never selects outliers but only items in the center of the feature space. Diversity sampling-all is the direct opposite of representativeness sampling and selects items that are as far from all other items as possible, and in doing so covers as much of the feature space as possible, with a tendency to select outliers.

Cluster Centroid Baseline

Another interesting baseline for comparison are classifiers trained on cluster centroids, as proposed by Zesch et al. (2015). Following their approach, we use Weka's k-means clustering to cluster the data, with k equal to the desired number of training instances. From each cluster, we extract the item closest to the centroid, build a training set from the extracted items, and learn a classifier from the training data. This process is repeated with varying numbers of training items: the first iteration has 20 labeled items, and we add in steps of 20 until reaching 200 labeled items. We then add data in steps of 50 until we reach 500 labeled items, and in steps of 100 until all data has been labeled. Note that this approach does not directly fit into the general AL framework. In AL, the set of labeled data is increased incrementally, while with this approach a larger training set is not necessarily a proper superset of a smaller training set but may contain different items.

Seed Selection

The seed set in AL is the initial set of labeled data used to train the first classifier and thus to initialize the item selection process. The quality of the seeds has been shown to play an important role for the performance of AL (Dligach and Palmer, 2011). Here we consider two ways of selecting seed set items.

First is the baseline of *(a) random seed selection*. Random selection can be suboptimal when it produces unbalanced seed sets, especially if one or more classes are not contained in the seed data at all or – in the worst case – the seed set contains only items of one class. Some of the ASAP data sets are very skewed (e.g., questions 5 and 6, see Table 6.2) and therefore carry a high risk of producing such suboptimal seeds via random selection.

The second condition is *(b) equal seed selection*, in which seed items are selected such that all classes are equally represented. We do this in an oracle-like condition, but presumably teachers could produce a balanced seed set without too much difficulty by scanning through a number of student responses. Of course, this procedure would require more effort than simply labeling randomly-selected responses.

The number of items in the seed set is another important AL parameter. While a larger seed set provides a more stable basis for learning, a smaller seed set shows benefits from AL at an earlier stage and requires less initial labeling effort. In the *small seed set* condition, and for both random and equal selection methods, 10 individual seed sets per prompt are chosen, each with either 3 or 4 seeds (corresponding to the number of classes per prompt). We repeat this process for the *large seed set* condition, this time selecting 20 items per seed set.

Batch Size

Batch size determines how many instances are labeled in each AL round. This parameter is especially relevant with the real-world application of SAS in mind. In real life, it may be inconvenient to have a teacher label just one instance per step, waiting in between labeling steps for retraining of the classifier.

On the other hand, sampling methods benefit from smaller batch sizes, as larger batches tend to contain a number of similar, potentially redundant instances. To combine the benefits of the two settings, we use *varying batch sizes*. To benefit from fine-grained sample selection, we start with a batch size of one and keep this until one hundred instances have been labeled. We then switch to a batch size of 5 until 300 instances have been labeled, and from then on label 20 instances per batch.

For comparison, we also run experiments where 20 instances are labeled in every AL step before a new classification model is learned, in order to investigate whether the potentially inconvenient process of training a new model after each individual human annotation step is really necessary.

6.1.4 Results

We now investigate to what extent active learning, using various settings, can reduce the amount of training data needed for SAS.

Evaluation of Active Learning

We evaluate all our SAS systems using Cohen's linearly weighted kappa (Cohen, 1968). Each result reported for a given combination of item selection and seed selection methods is the average over 10 runs, each with a different seed set. The seed sets remain fixed across conditions.

In order to evaluate the overall performance of an AL method, we need to measure the performance gain over a baseline. There a two options how this can be done, either by comparing performances of different methods after the same number of human annotation steps or by comparing the number of annotation steps necessary to reach a certain performances. We opt for the first one, as the performance curves tend to be non-monotonic, especially in the beginning. To further mitigate such effects, rather than comparing performances at one fixed point in the learning curve, we follow Melville and Mooney (2004) in looking at averaged performance over a set of points early in the learning curve. This is where AL produces the biggest gains; once many more items have been labeled, the differences between the systems reduces. We slightly adapt Melville and Mooney's method and compute the average percent error reduction (that is, error

reduction on kappa values) over the first 300 labeled instances (18-26% of all items, depending on the size of the data set).

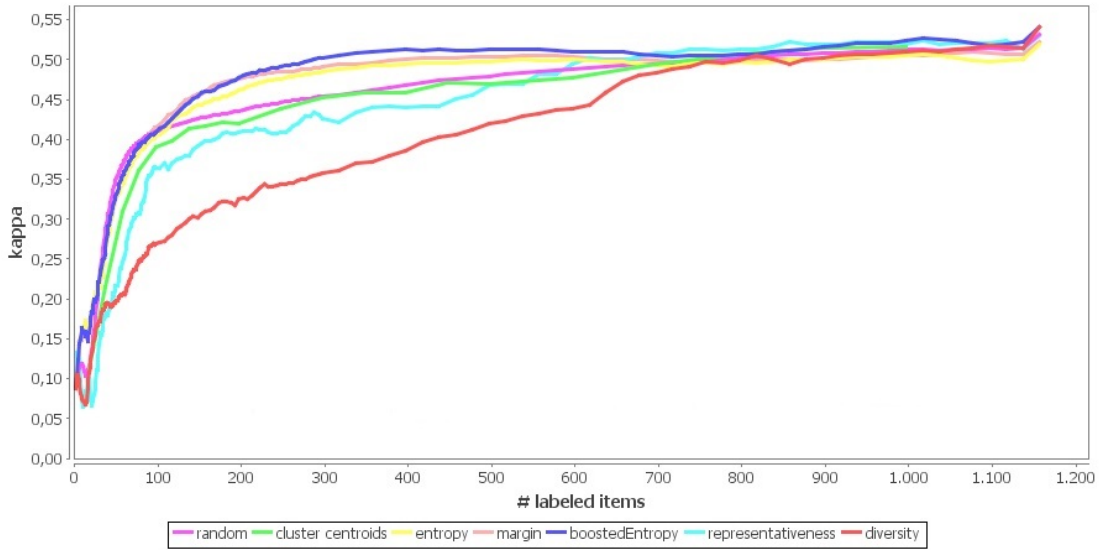

Figure 6.5: AL performance curves compared to two baselines: random item selection and cluster centroids. All results are averaged over all prompts and seed sets.

Experiment 1: Comparison of Different Item Selection Methods

The first experiment compares the different item selection methods outlined in Section 6.1.3, using small seedsets and varying batch sizes.

To give a global picture of differences between the methods, Figure 6.5 shows the learning curves for all sample selection methods, averaged over all prompt and seed sets. Especially in early parts of the learning curve until about 500 items are labeled, uncertainty-based methods show improvement over the random baseline. Both representativeness and diversity-based sampling perform far worse than random. On average, the systems trained on cluster centroids perform at or below the random baseline, which confirms the findings of Zesch et al. (2015) (though in a slightly different setting).

The picture changes a bit when we look at the performance of AL methods per prompt and with different seed selection methods. Table 6.6 shows the percent error reduction (compared to the random baseline) per prompt and seed selection method, averaged over the first 300 labeled items. Most noticeable is that we see a wide variety in the performance of the sample selection methods for the various prompts. For some - most pronouncedly prompt 2, 5, 6, and 10 - there is a consistent improvement for uncertainty sampling methods, while other prompts seem to be

almost completely resistant to AL. When looking at individual averaged AL curves, we can see some improvement for prompts 7 to 9 that peaks only after 300 items are labeled. For prompt 3, none of the AL methods ever beats the baseline, at any point in the learning process. We also observe variability in the performance across seed sets for one prompt, as can be seen from the standard deviation.

The question of which AL method is most effective for this task can be answered at least partially: if any method yields a substantial improvement, it is an uncertainty-based method. On average, boosted entropy gives the highest gains in both seed selection settings. Comparing random to equal seed selection, performance is rather consistently better when AL starts with a seed set that covers all classes equally.

Experiment 2: The Influence of Seeds

Experiment 1 shows a clear benefit for using equal rather than random seeds. In a real life scenario, however, balanced seed sets are harder to produce than purely random ones. One might argue that using a larger randomly-selected seed set increases the likelihood of covering all classes in the seed data and provides a better initialization for AL, without the additional overhead of creating balanced seed sets.

This motivates the next experiment, in which learning begins with seed sets of 20 randomly-selected labeled items, but otherwise follows the same procedure. We compare the performance of systems initialized with these larger seed sets to both random and equal small seed sets, considering only the more promising uncertainty-based item selection methods, and again using varying batch sizes.

Table 6.7 shows the results. We can see, that the performance for margin and entropy sampling is slightly better than the small random seed set (curiously not for boosted entropy), but it is still below that of the small equal seed set. However, the trend across items is not completely clear. We still take it as an indicator that seeds of good quality cannot be outweighed by quantity.

Experiment 3: The Influence of Batch Sizes

In Experiment 1, we used varying batch sizes that learn a new model after each individual labeled item in the beginning and allow larger batches only later in the AL process. In a real-life application, larger batch sizes might be in general preferable. Therefore, we test an alternative setup where we sample and label 20 items per batch before retraining.

Table 6.8 presents results for uncertainty-based sampling methods, averaged over the first 300 labeled instances. Compared to the varying batch size setup (numbers in parentheses), performance goes down, indicating that fine-grained sampling really does provide a benefit, especially

prompt & seeds	entropy	margin	boosted entropy	diversity	representativeness
1 Equal	-0.58 (5.8)	-0.05 (4.5)	-0.51 (4.0)	-30.53 (1.3)	-14.04 (2.8)
2 Equal	5.61 (5.1)	3.82 (7.4)	**6.75** (6.5)	-24.40 (0.5)	0.88 (1.7)
3 Equal	-2.42 (3.0)	-2.18 (5.1)	-2.32 (3.2)	-27.10 (0.9)	-11.34 (2.7)
4 Equal	-3.40 (7.5)	**1.44** (2.3)	-2.41 (6.6)	-14.67 (1.8)	-10.15 (5.8)
5 Equal	12.67 (2.5)	**15.38** (2.8)	12.25 (6.6)	-15.50 (2.7)	-9.44 (11.9)
6 Equal	21.49 (5.9)	22.70 (3.3)	**24.39** (2.6)	-16.47 (4.9)	-10.29 (3.5)
7 Equal	-1.49 (6.8)	-2.36 (6.4)	-2.97 (5.5)	-4.85 (1.4)	**0.65** (1.2)
8 Equal	-4.41 (8.6)	**0.26** (4.5)	-2.31 (5.3)	-9.71 (1.5)	-9.16 (4.3)
9 Equal	-2.91 (5.4)	-0.84 (9.1)	**3.32** (5.3)	-0.88 (5.5)	-9.10 (5.6)
10 Equal	7.97 (6.6)	8.33 (6.7)	**10.88** (6.3)	10.31 (3.7)	-4.92 (5.0)
avg	3.25 (5.7)	4.65 (5.2)	**4.71** (5.2)	-13.38 (2.4)	-7.69 (4.4)
1 Random	-4.24 (6.3)	-2.98 (8.0)	-0.33 (2.6)	-30.81 (2.2)	-13.10 (3.7)
2 Random	4.28 (5.7)	2.98 (7.6)	**6.14** (3.2)	-21.37 (1.1)	-0.82 (2.4)
3 Random	-11.41 (7.3)	-5.82 (7.3)	-5.52 (9.5)	-26.13 (2.6)	-11.13 (2.5)
4 Random	**0.18** (7.8)	-5.09 (9.8)	-1.73 (7.5)	-11.13 (2.2)	-11.11 (2.8)
5 Random	8.92 (5.0)	**12.93** (3.9)	10.86 (4.8)	-41.56 (16.0)	-2.20 (5.3)
6 Random	19.66 (3.9)	**21.13** (3.6)	19.29 (2.1)	-42.53 (26.6)	-11.41 (2.9)
7 Random	-4.21 (7.8)	0.39 (5.4)	-4.24 (7.6)	-4.22 (1.8)	**0.56** (2.3)
8 Random	-1.63 (7.3)	-0.52 (7.0)	-0.54 (4.3)	-10.19 (0.5)	-6.18 (3.7)
9 Random	-2.78 (6.9)	-4.35 (7.1)	-3.53 (6.3)	-3.17 (5.4)	-10.46 (6.1)
10 Random	4.89 (9.6)	7.74 (7.2)	**10.95** (5.0)	10.94 (3.4)	-3.01 (3.2)
avg	1.37 (6.7)	2.64 (6.7)	**3.13** (5.3)	-18.02 (6.2)	-6.89 (3.5)
all	2.31 (6.2)	3.65 (5.9)	**3.92** (5.2)	-15.70 (4.3)	-7.29 (4.0)

Table 6.6: Performance for each combination of prompt and seed selection method, reporting mean percentage error reduction on kappa values and SD compared to the random baseline.

Seeds	entropy	margin	boosted
Random – large seeds	1.45	2.72	2.57
Random – small seeds	1.36	2.63	3.12
Equal – small seeds	**3.25**	**4.65**	**4.71**

Table 6.7: Error reduction rates over random sampling for different seed set sizes, averaging over all prompts.

Seeds	entropy		margin		boosted	
Equal	-1.11	(3.25)	3.78	(4.65)	2.12	(4.71)
Random	0.04	(1.36)	2.60	(2.63)	0.93	(3.12)
All	-0.53	(2.30)	3.19	(3.64)	1.53	(3.92)

Table 6.8: Error reduction rates over random sampling for large batch size and small seed sets, averaging over all prompts. Scores from the varying batch size setup appear in parentheses.

early in the learning process. Where larger batch sizes may lead to selection of instances in the same region of uncertainty, a smaller batch size allows the system to resolve a certain region of uncertainty with fewer labeled training instances.

6.1.5 Variability of Results across Data Sets

On average, it is clear that uncertainty-based active learning methods are able to provide an advantage in classification performance over random or cluster-centroid baselines. If we look at the result for the different prompts, though, it is equally clear that AL performance varies tremendously across data sets for individual prompts.

In order to deploy AL effectively for ASAS, we need to better understand *why* AL works so much better for some data sets than for others.

In Table 6.6 we see that AL is especially effective for prompts 5 and 6. Cross-referencing Table 6.2, it becomes clear that these are the two ASAP prompts with the highest degree of class imbalance. Figure 6.9 shows the changes in the distribution of the individual classes among the labeled data for prompt 6 as AL (here with entropy item selection) proceeds. We see clearly that uncertainty sampling at early stages selects the different classes in a way that is more balanced than the overall distribution for the full data set and thus increases the accuracy of the classifier in labeling minority class items. For comparison, a plot for random sampling would ideally consist of four lines parallel to the x-axis, and both diversity and representativeness sampling tend to select items from the majority class, explaining their bad performance.

Class imbalance explains some of the variable performance of AL across prompts, but clearly there is more to the story. Next, we use language model (LM) perplexity (computed using the SRILM toolkit (Stolcke, 2002)) as a measurement of how similar the classes within a prompt are to one another. We measure this per class by training a LM on the items from all other classes (for the same prompt) and then compute the average perplexity of the target class items under

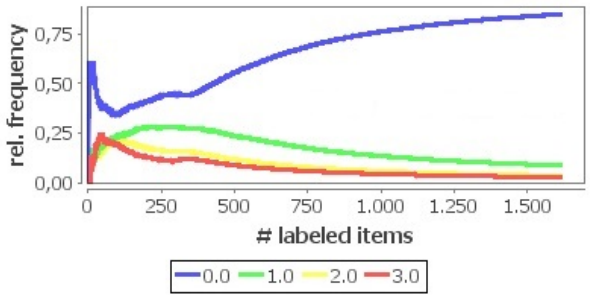

Figure 6.9: Distribution of individual classes among the labeled data for prompt 6, using entropy sampling.

the "other-classes" LM. Higher average perplexity means that the items in the class are more readily separable from items in other classes.

prompt	score 0.0	score 1.0	score 2.0	score 3.0
1	156	46	27	45
2	104	48	52	56
3	44	23	64	-
4	78	59	55	-
5	970	88	52	49
6	907	76	60	44
7	338	117	45	-
8	535	70	47	-
9	633	127	56	-
10	304	49	39	-

Table 6.10: Average perplexity per prompt and class under LMs trained on all "other-class" items from the same prompt.

Table 6.10 shows the results. We see that for those answers that work well under AL, again prominently prompts 5 and 6, at least some classes separate very well against the other classes. They show a high average perplexity, indicating that the answer is not well modeled by other answers with different scores. In comparison, for some other data sets where the uncertainty curves do not clearly beat random sampling, especially 3 and 4, we see that the classes are not well separated from each other. They are among those with the lowest perplexity across scores.

This result, while preliminary and dependent on knowing the true scores of the data, suggests that uncertainty sampling profits from classes that are well-separated from one another, such

that clear regions of uncertainty can emerge. An intriguing future direction is to seek out other approaches to characterizing unlabeled data sets, in order to determine: (a) whether AL is a suitable strategy for workload reduction, and (b) if so, which AL setting will give the strongest performance gains for the data set at hand.

6.1.6 Conclusions

In this study, we have investigated the applicability of AL methods to the task of SAS on the ASAP corpus. Although the performance varies considerably from prompt to prompt, on average we find that **uncertainty-based sample selection** methods outperform both a random baseline and a cluster centroid baseline, given the same number of labeled instances. Other sample selection methods capturing diversity and representativeness perform well below the baselines.

In terms of seed selection, there is a clear benefit from an **equal seed set**, one that covers all classes equally. A small equal seed set is preferable even to a larger but potentially unbalanced seed set. In addition, we see benefits from a **variable batch size** setting over using a larger batch size. It is beneficial to proceed in small steps at the beginning of learning, selecting one item per run, and only move to larger batch sizes later on.

We see two interesting avenues for future work. First, the influence of the quality of seed set items with respect to the coverage of classes raises the question of how best to select – or even generate – equally distributed seed sets. One might argue whether an automated approach is necessary: perhaps an experienced teacher could easily browse through the data in a time-efficient way to select clear examples of low-, mid-, and high-scoring answers as seeds.

The second question is the more challenging and more important one. The variability of AL performance across prompts clearly and strongly points to the need for better understanding how attributes of data sets affect the outcome of AL methods. A solution for predicting which AL settings are suitable for a given data set is an open problem for AL in general. Further steps in this direction need to be taken before AL can be reliably and efficiently deployed in real life assessment scenarios.

6.2 Clustering Study 1: Finding a Trade-off between Accuracy and Rater's Workload in Grading Clustered Short Answers

This study investigates the potential of answer clustering and label propagation as an approach to semi-automatic scoring in the domain of foreign language learning. In particular, we explore the trade-off between grading accuracy and reduction of teacher workload, asking: can we achieve

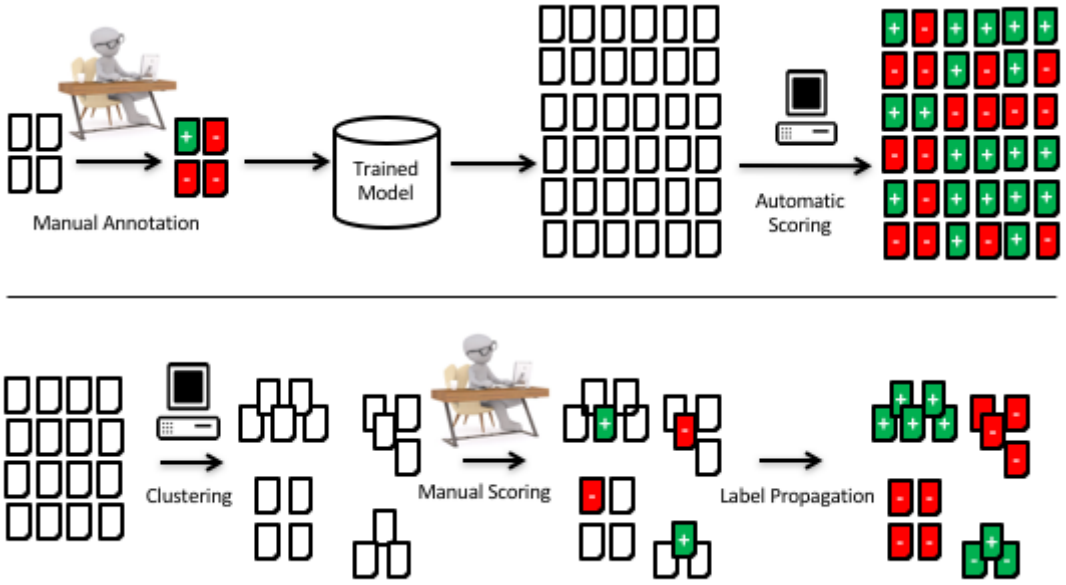

Figure 6.11: Clustering with label propagation (lower half) in contrast to a classical supervised machine learning-based approach (upper half). Visualization inspired by Zesch et al. (2015)

a significant reduction in the number of items a teacher needs to grade while maintaining an acceptable scoring accuracy?

The study focuses on scoring of answers to short answer questions (usually one or two sentences) to listening comprehension exercises for learners of German as a foreign language. With this study we move away from supervised scoring models and consider a real-life scenario for manually grading short answers.

Our approach is based on the assumption that highly similar student answers are likely to receive the same grade from a teacher and thus can be grouped and graded as a single unit. To do this, we use clustering techniques based on surface features (similar to, e.g., topic clustering as described in Steinbach et al. (2000)). We simulate a grading scenario in which answers are clustered automatically, teachers label only one item per cluster, and that label is then propagated to the other items in the same cluster.

Figure 6.11 illustrates our clustering approach in comparison to the supervised machine learning approaches used in Chapter 5 and in the active learning study in Section 6.1.

In listening comprehension, the most important factor for identifying correct answers is semantic content; minor errors in spelling or grammar do not lead to answers being labeled as

incorrect. Answers with high lexical overlap are likely to receive the same grade from a teacher. At the same time, a high spelling variance can be observed in listening comprehension data more than in reading comprehension, as students tend to write what they hear and cannot rely on a reading text to copy lexical material from. Both the high orthographic variance and the fact that it is to be ignore for scoring supports the use of surface-level clustering.

A similar approach, developed in parallel by Basu et al. (2013), targets the grading task for short answer questions by forming clusters and evaluating the number of human actions needed to correct a set of answers. In this context, actions consist of labeling complete (sub-)clusters of answers.

We evaluate our approach directly on data from placement tests for learners of German as a foreign language at Saarland University. For this study, we work under the assumption that it is a tolerable outcome to have a small number of incorrectly-graded answers. This particularly fits the placement testing scenario, where the aim is to determine a base language level for students, and where the listening comprehension component is only one out of several parts of the examination. The scores from all components (which include item types that are more easily automatically graded, such as multiple choice or fill-in-the-blank exercises) are combined to come to a final placement for the student, who receives only an aggregated score and no indication of performance on individual items. It must be noted that this is as yet an untested assumption, and for high-stakes testing our tolerable-amount-of-error assumption would not hold.

Goals of the Study

This study addresses research question 2.3:

RQ 2.3: Can clustering reduce a teacher's workload in an assisted scoring scenario?

We address this question by clustering answers based on surface features and using different label propagation techniques that simulate how the clustering can be integrated in real-life scoring scenario.

Contributions

Under the above-made assumption that a number of misclassified items is acceptable in a placement testing scenario, our clustering experiments show promising results. The system can achieve scoring accuracy of 85% or above (depending on precise system settings) when teachers label only 40% of learner answers.

6.2.1 Data

For this study, we use the Laempel listening comprehension data as described in Section 3.4. One way data from listening comprehension tasks differs from reading comprehension data is the higher frequency of spelling errors. When answering reading comprehension questions, it is well known that language learners often directly copy relevant material from the text into their answers. This strategy is known as lifting in the second language acquisition community. Lifting leads to a high overlap of both lexical material and orthography between the text and the learner answer. Learners answering listening comprehension questions make less frequent use of material from the audio input; this leads to a much higher degree of orthographic variability in learner answers. To determine the extent of the variability, we run the data through a German spelling correction system[2], with the result that 18.6% of words are unknown. Our clustering approach needs to account for this variability. Another difference between the listening comprehension data we use and typical short answer settings is length. Where most work on short answer scoring re- ports answers that are 1 -3 sentences in length, the average length of learner answers in the data we use is 4.8 tokens, not counting punctuation.

We use all 1777 individual non-empty answers to 21 different questions, collected from 98 students. We treat each grade (typically 0.0, 0.5, and 1.0, for some answers up to 2.0 points) as an individual, discrete label.

6.2.2 Features and Modeling

As punctuation or capitalization errors are irrelevant in the scoring of short answer questions, we remove sentence punctuation and lowercase all learner answers. After these preprocessing steps, we merge string-identical answers. This step already reduces the number of different items to be graded by 25%. In this way we build a set of answer types based on string identity (e.g., *"in Berlin"* is a different answer type from *"in the north of Berlin"*, but the same as *"in berlin"*). All further clustering is done on answer types, and all reported statistics are also over these types. We extract features based on word n-grams, character n-grams and keywords. For word feature extraction we first lemmatize all words using Treetagger (Schmid, 1994). We then extract word uni-, bi- and trigrams as well as skip-bi-, and trigrams (i.e., pairs and triples of words with an arbitrary number of other words in between).

To handle spelling errors, we use character bi- to four- grams extracted from the unlemmatized text. For example, the two answers in 6.1 below are clearly conveying the same material, but if we only consider word overlap in clustering, the only shared lemma would be the pronoun she, which is unlikely to result in the two answers sharing a cluster at any meaningful level of

[2]The German version of aspell `http://aspell.net/`

clustering. Character n-grams, on the other hand, are able to capture such similarities.

(6.1) *She lives in Berlin*
She livs Berlim

We decide against correcting learner answers with a spellchecker in order to allow for the case that a mis- spelled form is a misunderstanding rather than an orthographic mistake.

QM Condition. As an optional preprocessing step we exclude from the answer strings lexical mate- rial contained in the question. In this condition, we want to treat as equivalent answers which reiterate the theme of the question and those which simply state the rheme. For example, for the question *"Where does she live?"* the two answers seen in 6.2 would be treated as belonging to the same answer type.

(6.2) *She lives in Berlin*
Berlin

We refer to this as the exclude-question-material option (QM+).

KEY Condition. Finally, we define keywords for each question based on the target answer given by the teachers. These specify the minimal requirement of lexemes that should be present in a correct answer and consist mainly of the nouns in the target answers. Consider the following question and target answer pair:

(6.3) Q: *Why does she have to leave?*
A: *Ihr Deutschkurs beginnt bald.*
Her German language course starts soon.

In this case, there is one relevant keyword: *"Deutschkurs"*; for English data we would include the two phrases *"German course"* and *"German language course"*. Some orthographic variation is allowed for detecting keywords in learner answers. The KEY feature is implemented by determining, for each keyword, whether or not it is present within an answer type. To give this feature greater weight, it is repeated 100 times in the feature vectors.

6.2.3 Clustering and Experiments

In this section we discuss parameters and evaluations used for clustering, followed by presentation of experiments and experimental results.

Clustering and Evaluation Metrics

We use single-pass clustering. For each item, we calculate cosine similarity between its feature vector and the centroid of each existing cluster. If the highest similarity value is above the specified threshold, we add the item to the cluster whose centroid it is most similar to. Otherwise we establish a new cluster based on the item in question. Clustering is done individually per question.

We consider four conditions resulting from two binary options for feature extraction: whether or not to use the keyword feature (KEY+ and KEY-) and the exclude-question-material option (QM+ and QM-). We also vary the similarity threshold from 0.0 to 1.0 (in steps of 0.1). At one extreme, using a similarity threshold of 0.0 results in all items being placed in a single cluster. At the other extreme, enforcing a threshold of 1.0 means that each items ends up in its own single-item cluster.

In addition to standard cluster evaluation metrics as described in Amigó et al. (2009), we want to assess the usefulness of clustering to a teacher in a simulated grading scenario, where a teacher grades just one item per cluster. Therefore, we compute the accuracy achieved if the label assigned by a teacher to this one exemplary item is propagated to all other items in the cluster. The number of clusters is then a suitable approximation of teacher workload.

Here we consider three different conditions with respect to selecting the item to be graded. As a baseline, we randomly select one answer type from each cluster. In a second, more informed, method, we select the single answer type that is closest to the centroid of the cluster. This method aims to choose for labeling the item that is most representative of its cluster. Finally, to simulate the best possible grading accuracy given a particular clustering, we explore a third option, an oracle condition in which we assume that we are always able to select an item from the majority class of a cluster. This measure represents best reachable grading accuracy for a given clustering. We compute accuracies for these three conditions for different thresholds, with those different thresholds leading to different numbers of clusters. In order to avoid effects from applying the single-pass clustering on items in a particular order, we run the clustering always 20 times on different random orderings of our items. Unless indicated otherwise, we report all results as the average over these 20 runs. Note that clusters are re-computed for each threshold and are thus not hierarchically organized. That means that while a higher threshold leads to a higher number of clusters and thus more items being labeled by a teacher, it does not necessarily always lead to a higher grading accuracy. For a few noisy items, it can happen that having more clusters actually leads to a decrease in accuracy.

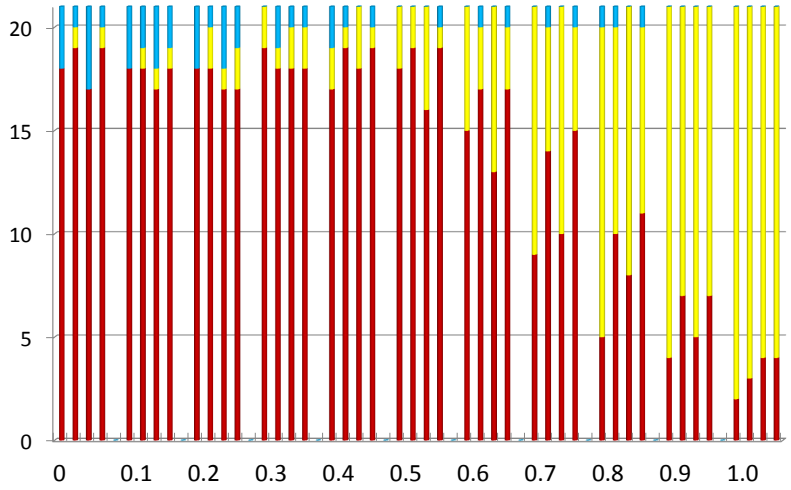

Figure 6.12: This graph shows, for each threshold, the relative performances of our two item selection methods. For the 21 questions in our data set, we indicated how often centroid-based selection is better than random (bottom/red), how often they are equally good (middle/yellow), and how often random selection does better (top/blue). The four columns for each threshold are (left to right): KEY- QM-, KEY+ QM-, KEY- QM+, KEY+ QM+

Experiment 1: Comparison of Item Selection Methods

The first experiment compares methods for selecting the one-item-per-cluster to be labeled. We compare *random item selection* to *centroid-based item selection*.

We see (Figure 6.12) that in most conditions, scoring based on centroid-based item selection leads to a higher accuracy than random selection. For higher clustering thresholds, the random and centroid-based selection are often equally good, because there are more clusters with fewer items. This has the effect that both random and informed selection more frequently result in the majority class label (trivially so for single-item clusters).

Figure 6.13 shows the magnitude of this performance difference, averaging over questions. Different styles of boxes are used for the four different conditions. We see that with lower similarity thresholds, there is a greater increase in performance from using informed item selection.

Figure 6.14 compares centroid-based scoring to the oracle condition in which we score each cluster according to the majority class label. For each threshold, we plot the number of questions (out of 21) for which centroid-based scoring reaches the same accuracy we achieve if we score according to the majority class. The general trend across thresholds and clustering conditions is that for more than 15 of the 21 questions, centroid-based item selection does as well as the oracle condition.

Experiment 2: Reducing Workload while Keeping an Acceptable Accuracy

This second experiment is motivated by the assumption that scoring with some degree of error (90% accuracy) would still be useful to a teacher in a real-life scenario, provided it comes with a significant reduction in workload. We evaluated exemplarily on one run of our clustering experiments, how many answer types need to be labeled in order to obtain this accuracy. The main result for our scenario is that, averaging over questions, this accuracy can be achieved by labeling only 40% of all answer types using centroid-based item selection.

However, the scoring accuracy achieved with this amount of labeled data varies considerably from question to question. Table 6.15 shows the range of variation for one exemplary experimental setting (centroid-based selection, similarity threshold of 0.4, KEY+, QM+). We see, for example, that for 5 questions, labeling maximally 20% of answer types is sufficient to reach 90% accuracy; for another 5 questions, upward of 80% of answer types need to be labeled. It is clear that question type plays a role in the effectiveness of such semi-automatic scoring strategies; further exploration of the influence of question type is needed if such a strategy is to be successfully implemented in a non-simulation setting.

Putting the influence of question types to the side, we ask what would actually happen if we were to apply this 40% labeling strategy to all questions in our placement test scenario; results are shown in Table 6.16. First, we measure the overall scoring accuracy if, for each question, we apply our centroid-based scoring strategy to the clustering produced by the lowest threshold that results in labeling 40% or more of the answer types. Next, we choose instead the *highest* threshold that leads to 40% *or less* of the answer types being labeled.

We see here that, despite differences due to question type, a labeling accuracy between 85 and 90% (depending on settings for KEY and QM) can be achieved by labeling only 40% of answer types. We do not observe interesting differences between the four individual conditions.

6.2.4 Conclusions and Future Work

We have shown in this work that an answer clustering and label propagation strategy can be used to reduce a teacher's grading workload while still maintaining a grading accuracy near 90%. We

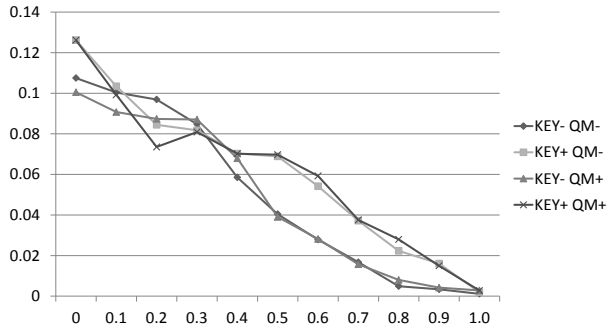

Figure 6.13: Accuracy gain if we use centroid-based instead of random item selection

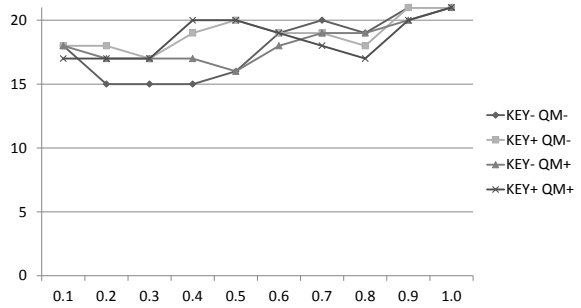

Figure 6.14: How often does the centroid-based item selection reach the same accuracy as the oracle condition?

percentage of answer types needed for 90% accuracy	#questions within that range
0 -19.99	5
20-39.99	7
40-59.99	3
60-79.99	1
80-100	5

Table 6.15: Variety of data (in percent of the answer types for one question) needed to get 90% accuracy

	40% or more	40% or less
KEY-, QM	0.907	0.844
KEY+, QM	0.897	0.851
KEY-, QM +	0.915	0.870
KEY+, QM +	0.897	0.852

Table 6.16: Accuracies obtained when choosing a threshold so that either at least or at most 40 % of the data is labeled

have shown this using learner answers to German short answer questions, simulating a strategy of labeling just one item out of each cluster.

In the next study, we will investigate how to further improve the quality of the clustering by semi-supervised clustering methods.

Another important question to investigate is the so far untested assumption that a grading accuracy of 90% is a useful and acceptable accuracy for certain grading scenarios. As a first approximation, we could say that, if errors are evenly distributed over all students and all test items, and the likelihood of misgrading any individual item is 10% for each of the roughly 20 items on the test, the probability of mis-scoring 4 or more items for any single student can be estimated at below 15%. We calculate this using a Bernoulli distribution to determine for what percentage of students the test produces at most a certain number of errors. However, it is clear that some items are harder than others, and the effectiveness of our semi-automatic scoring strategy varies considerably across questions.

Factors that seem to contribute to this are the expected length of the answer, the type of information asked for (e.g., just one NP or a whole sentence) and the type of question (e.g., reproduction or inference). One important step for future work will be to explore how to make use of those properties in clustering settings.

Another point for future work should be the detection of elements within a cluster that have the minority label of this cluster, and thus would get the wrong label even under a well-performing cluster-label-propagate strategy. This could be done, for example, by understanding teachers thresholds between true misspellings and cases of similar words with different semantics better. For example for a target answer of *"Angola"*, some spelling variants (*"Angla"*, *"Engloa"*, etc.) may be acceptable, while words like *"England"* or maybe even proper names like *"Angelo"* might not, despite containing similar character n-grams).

6.3 Clustering Study 2: Using Semi-Supervised Clustering for Short-Answer Scoring

In the previous two studies in this chapter, we investigated how to best invest human annotation effort by either labeling items for supervised classification or to serve for label propagation in clustering. With the study in this section, we re-address the question how best to use a teachers available time, but we combine the advantages of clustering and classification by using semi-supervised clustering.

Most approaches to ASAS consider automatic scoring as a classification task, relying on supervised machine learning (ML) techniques which require manually labeled training data. In contrast, our clustering study in the previous section, as well as Basu et al. (2013) and Brooks et al. (2014) focuses on the use of clustering techniques for ASAS (see Section 2.4.1).

Some amount of human scoring is required for both supervised ML and clustering: annotation of training data in the one case, and annotation of representative cluster members as a basis for propagation in the other case. Zesch et al. (2015) compared the performance of clustering with that of ML methods, keeping the number of manually labeled items constant. They carried out their study on the PG data set and in addition on the ASAP data set (see below, Section 6.2.1). They reported that clustering proved beneficial only on the short and simple answers (a few words) of the PG data set. On the ASAP data set with more complex, longer answers, clustering falls far behind ML methods in their experiments.

In this study, we want re-address the question whether clustering and label propagation can help in the ASAS task. In particular, our approach addresses the following shortcoming of (Zesch et al., 2015)

- Clustering in (Zesch et al., 2015) is always performed on the full feature set (various n-gram and dependency features), that is also used for the supervised ML. While ML algorithms typically learn which features are more important than others, standard clustering algorithms do not do that by themselves. Therefore, we use methods of attribute selection and dimensionality reduction to make sure that only relevant features are used for clustering.

- In (Zesch et al., 2015), human annotations are used in the setup in exactly one way: to label cluster centroids as the basis for label propagation. Instead, our method investigates several ways how human annotation efforts can additionally be used to create constraints for the clustering process, learn about the quality of features and contribute evidence for how to label a cluster.

6.3.1 Goals of this Study

With this study we contribute further to answering research question

RQ 2.3: Can clustering reduce a teacher's workload in an assisted scoring scenario?

In contrast to our previous study, we address whether extending unsupervised clustering with different kinds of supervision is beneficial for scoring performance. More precisely, we investigate whether *semi-supervised clustering* can improve clustering results.

While existing clustering approaches use manually labeled data only for post-clustering label propagation, we distribute human effort and use human-labeled data in multiple ways before, during and after clustering:

- **Feature selection:** We use labeled items for feature selection *before* the actual clustering, as clustering algorithms are known to suffer more from noisy features than supervised learning algorithms that can select the features relevant for a task (Alelyani et al., 2013).
- **Clustering with constraints:** we employ two methods of using labeled instances as *seeds during* the clustering: (i) for guiding the clustering process through relational constraints that indicate whether two instances cannot or must belong to the same cluster, and (ii) for metric learning, i.e., adapting the distance metric according to those constraints. We reuse the items labeled for feature selection as seeds, so the second step does not require additional annotation effort.
- **Label propagation**: we use label propagation *after* clustering to assign a label to each cluster based on the teacher-assigned label of just one item of the cluster using a propagation method described in the previous study. We annotate the item closest to the centroid and propagate its label to all cluster members, as this procedure selects prototypical instances and is superior to propagating a random label.

6.3.2 Contributions

This study shows that *semi-supervised clustering* can substantially improve clustering results. The multi-purpose usage of labeled instances can overcome part of the gap between clustering and supervised ML methods, not only for the short answers from PG, but also for the complex ASAP data set. We investigate scenarios with different amounts of available human annotation steps as well as different trade-offs for allocating the human annotation effort to the three different annotation steps described above.

We conclude that clustering with label propagation can be an alternative to supervised ML methods, since it has the advantage of providing teachers with structured sets of answers.

6.3.3 Method

Data Sets

We run experiments on two data sets, ASAP (see Section 3.3) as a representative of a data set with relatively long answers (between 26 and 66 tokens on average per prompt) and Powergrading (see Section 3.5)) with very short answers, typically consisting of only a few words.

Features and Feature Selection

In congruence with previous work, we concentrate mainly on lexical features as they are highly predictive for this task. In the Kaggle competition for the ASAP data set, the top 5 best-performing systems used mainly lexical features for scoring (Higgins et al., 2014) (the best-performing system (Tandalla, 2012) was indeed one that additionally used hand-crafted regular expressions for each prompt). For the more complex answers of ASAP, which usually consist of complete sentences instead of short phrases, we use lemma and character n-grams, and dependency subtrees as features. For the PG data set with very short phrasal answers, dependency parsing provided unsatisfying results, so we restricted ourselves to character and word n-gram features.

We use the labeled seeds to perform supervised feature selection, as clustering is particularly sensitive to noisy features (Alelyani et al., 2013).

We use Weka's information gain-based attribute selection and test different numbers of features including the full feature set. For most prompts, we reach optimal clustering results with either 200 or 100 features. We use the optimal size of feature sets per prompt in all experiments for both clustering and supervised ML. We tried other linear feature selection algorithms but found no significant differences in performance. We also explored subset evaluation as an alternative, using Weka's Cfs Subset evaluation, and found it to be less suitable than Information Gain.

Semi-supervised Clustering

Clustering algorithms aim at grouping similar objects together, where similarity is measured by a distance metric. Standard clustering algorithms work completely unsupervised, only based on the distance metric. Semi-supervised clustering makes use of seed data gained through human annotation. Seed data can either be given in the form of labeled items expressing cluster membership, or as relational information stating that two items should or should not belong to the same cluster.

In our ASAS scenario, we assume that there is a one-to-many rather than a one-to-one relation

between scores and clusters. That means one score (out of the maximum of 4 different scores for the ASAP data set) can contain answers that fall into different groups of semantically similar answers. Especially for low-scoring answers there is certainly more than one way to "get it wrong", and thus we cluster into more clusters than there are labels. Answers with different scores should definitely go into different clusters, answers with the same score may or may not belong to the same cluster, dependent on their semantic relatedness. Therefore, we cannot use categorical seed information to estimate the number of clusters and to initialize seed clusters, but have to use relational information. Since scoring of individual answers is a much more natural task for teachers than assessing the similarity between different answers, we derive the relational pairwise constraints required for semi-supervised clustering from individually labeled items, the seeds. More specifically, we create a *cannot link (CL) constraint* stating that two answers should not go into the same cluster for each pair of seeds with different scores. In general, semi-supervised clustering can also use *must link constraints* stating that two items belong to the same cluster. We cannot derive reliable must link information from answer scores, so we employ cannot links only.

Implementation Setup We use the Weka implementation (Hall et al., 2009) of the unsupervised k-means algorithm (KM) (Lloyd, 1982) as our baseline algorithm, as do Zesch et al. (2015): k-means minimizes an objective function that sums over the squared distances of each item to its cluster centroid. As distance metric, we use Euclidian distance between feature vectors.

For semi-supervised clustering, we use extensions of k-means introduced in the *metric pairwise constrained k-means* (MPCKM) algorithm by Bilenko et al. (2004), who integrate the usage of pairwise constraints and metric learning into the k-means algorithm and provide an extension of the Weka API for that.[3] Constraints are integrated into the clustering in the form of penalties for constraint violations that are added to the objective function. Each constraint is associated with an importance weight.

Metric learning is done in the MPCKM algorithm after each k-means iteration by adjusting the weights of individual features in two ways: first, by moving existing clusters from the previous iteration further away from each other and second, by increasing the distance between items with violated CL constraints.

Label Propagation

For our experiments, we assume the following scenario: a teacher is given one item per cluster for scoring, and the score is propagated to all members of the cluster. Accordingly, we evaluate

[3] http://www.cs.utexas.edu/users/ml/risc/

our experiments using *label propagation* following both Horbach et al. (2014a) and Zesch et al. (2015).

We use **centroid propagation** as a realistic method, where the label for all answers in a cluster is based on just one labeled instance. We select for labeling an item which is prototypical for its cluster by selecting the one closest to the cluster centroid.

We consider **majority propagation** to provide an upper bound of performance that we could reach when labeling a cluster based on the label of one instance: we reach the best possible score for a given cluster if the one element whose label is propagated belonged to the majority class for that cluster. This evaluation is an oracle condition that indicates the quality and potential of a given clustering, as there is no reliable way to automatically select such an element.

Treatment of Duplicate Items in Clustering

The PG data set contains high numbers of duplicate answers; there are 2434 unique answer for a total of 6980 individual answers. Multiple annotation of duplicates does not add any information. Hence, we make sure that we never select duplicates when sampling answers for human annotation. However, the negative impact of performance is higher if we get a very frequent answer wrong compared to erring on an answer that is only given by one student. Therefore we do not remove duplicates when clustering or classifying the answers, such that multiple occurrences of an answer have more influence in the clustering process, as they have, e.g., a higher probability to be selected for centroid-based label propagation.

Baselines

We compare our clustering results to two baselines: *unsupervised k-means clustering* on the full feature set and *supervised ML*. To enable meaningful comparison between the methods, we keep the number of annotated instances n constant across all experimental conditions. Thus we create n clusters in unsupervised k-means clustering as all human annotation effort can be used to label cluster centroids. We have fewer clusters in the semi-supervised case, where some annotations are used for labeling seeds instead of cluster centroids. Accordingly, the ML baseline, implemented by Weka's SMO algorithm (Hall et al., 2009), is trained on n labeled items, as done in Zesch et al. (2015).

The baseline used in Zesch et al. (2015) is supervised ML with the complete feature set. We find that this baseline gives clustering with feature optimization an unfair advantage, as supervised ML algorithms also profit from feature selection. Thus we use an additional baseline, performing feature selection and report results for the best configuration per prompt. We randomly sample the data for the classifier 100 times and report average results. For the optimized

feature set we also report the best individual run as an upper bound.

6.3.4 Experiments

Our experiments address the question how a set of answers can be optimally graded with only a limited amount of available human annotation effort:

Experiment 1 compares variants of the k-means algorithm that correspond to different degrees of supervision to confirm the contribution of the individual components of the MPCKM algorithm. Experiment 2 investigates the optimal tradeoff for distributing a given amount of human annotations between (a) labeling seeds before clustering and (b) labeling cluster centroids after clustering. In Experiment 3 we cross-check that our semi-supervised results cannot be reached with approaches that use unsupervised feature selection. In Experiment 4 we investigate, how human annotation effort can be further minimized, by reusing seeds for label propagation.

Experimental Setup

Data Set Sizes In order to evaluate always on the same number of answers per data set, we use the first 1000 answers to each ASAP prompt, and all 698 answers to each PG prompt.

Evaluation Metric We report Cohen's quadratically weighted kappa (Cohen, 1968) after label propagation. In our grading scenario, where we want to measure the quality of the resulting grading of a set of answers and compare to supervised classification methods, this type of evaluation is more meaningful than evaluation measures applied in other clustering tasks, such as the widely used bCubed metric (Amigó et al., 2009).

Experiment 1: Different Degrees of Supervision in Clustering

In our first experiment, we measure the influence of different levels of supervision. We go from unsupervised k-means clustering (KM_{all}), over k-means clustering that uses seeds only for feature selection (KM_{sel}) and semi-supervised clustering that additionally derives CL constraints from the seeds (*KMCL*) to the full *MPCKM* clustering algorithm with feature selection, CL constraints and metric learning. We aim at investigating the effect of a fixed "small" number n of labeled data on clustering performance, which at the same time should be large enough to induce clusters of reasonable quality. We decided for n = 150 (out of a total of 1,000 answers per question) for ASAP, and n = 50 for PG (the comparably low number is due to the high amount of duplicates in the answers). This overall number of annotation steps is split into those answers that are used for both feature selection and constraints (the seeds), and those that are used to

	Clustering 40 cluster, 110 seeds					supervised ML 150 items		
p.	KM_{all}	KM_{sel}	KMCL	MPCKM	*$MPCKM_{best}$*	ML_{all}	ML_{sel}	ML_{best}
1	0.462	0.541	0.547	0.593	*0.668*	0.651	0.673	*0.711*
2	0.432	0.47	0.469	0.496	*0.574*	0.571	0.571	*0.62*
3	0.343	0.378	0.377	0.379	*0.451*	0.384	0.384	*0.437*
4	0.543	0.547	0.549	0.581	*0.651*	0.639	0.655	*0.693*
5	0.617	0.622	0.631	0.69	*0.756*	0.68	0.72	*0.782*
6	0.682	0.646	0.64	0.74	*0.765*	0.692	0.745	*0.787*
7	0.352	0.398	0.402	0.447	*0.533*	0.565	0.565	*0.622*
8	0.448	0.44	0.437	0.471	*0.556*	0.553	0.566	*0.61*
9	0.546	0.564	0.567	0.61	*0.686*	0.647	0.66	*0.698*
10	0.614	0.614	0.614	0.651	*0.715*	0.629	0.684	*0.738*
avg	0.5039	0.522	0.5233	0.5658	*0.6355*	0.6011	0.6223	*0.6698*

Table 6.17: Result on the ASAP data set

	Clustering 10 cluster. 40 seeds					supervised ML 50 items		
p.	KM_{all}	KM_{sel}	KMCL	MPCKM	*$MPCKM_{best}$*	ML_{all}	ML_{sel}	ML_{best}
avg	0.7928	0.7244	0.7493	0.7695	0.7864	0.7001	0.7213	0.8848

Table 6.18: Average results on the PG data set

label the centroid of each cluster; i.e., the number of annotations for labeling clusters centroids determines the number of clusters created.

In this experiment, the split of the n labeled items between seeds and labeled cluster centroids is 110:40 for ASAP and 40:10 for PG. These proportions of seeds and cluster centroids are selected as the optimal ones, based on the results of Experiment 2 (see below).

Tables 6.17 and 6.18 show the results for the different k-means variants. In addition to centroid-based label propagation, we report majority propagation $MPCKM_{best}$ for the full MPCKM algorithm as an upper bound for clustering performance.

We can learn the following from the experiment: first, investing labeled items into feature selection pays off (KM_{all} vs KM_{sel}) for the ASAP data set. Second, we see that adding constraints alone gives us an additional small improvement (KMCL vs KM_{sel}), while adding metric learning (MPCKM) adds substantially to the performance. The improvement is consistent for centroid-based label propagation and for the majority propagation upper bound $MPCKM_{best}$. Third, we see that the best clustering method comes closer to the ML baseline trained on the full data set (ML_{all}). By making optimal use of the manually labeled data, we could thus close more then half of the gap between the performance of clustering and machine learning stated in

Zesch et al. (2015).

For the PG data set, basic clustering already outperforms ML methods, arguably because the very short answers of the PG data set yield an already small feature set that contains little noise. For the following experiments, we therefore report results on the more challenging ASAP data set only. Note that our scores for ASAP are not directly comparable to the scores of the top performing systems from the Kaggle competition, as the evaluation setup, especially the number of training and test instances used, is different.

Experiment 2: Finding a Tradeoff between the Numbers of Seeds and the Number of Clusters

In this experiment, we determine the optimal tradeoff between the number of seeds, which are used for feature selection and constraints before clustering, and the number of clusters, where the centroid of each cluster is labeled after clustering. To do so, we evaluate the effect of different splits between labeled seeds and cluster centroids for MPCKM clustering with centroid-based label propagation, for different sizes of n using linearly weighted kappa. The results obtained for n = 100, 150, 200, 250 are shown in the curves of Table 6.19; plotted on the x-axis is the percentage of annotation steps used as seeds. The curves cover distributions from 0 seeds (n clusters) to n-10 seeds (10 clusters).

Unsurprisingly, we see that a higher overall number of annotated data yields a better clustering performance. As an interesting result of the experiment, we observe that the curves peak always between 75 and 80% of annotated data used as seeds, i.e., we profit more from adding more seeds then from adding more clusters.

Experiment 3: Comparison with Unsupervised Dimensionality Reduction

In Experiment 1, we used labeled seeds for supervised attribute selection. The clustering literature, however, also proposes unsupervised dimensionality reduction methods (Alelyani et al., 2013). Since this might have a similar effect without using any seeds, we compare our results on supervised feature selection from Experiment 1 to two methods of unsupervised feature selection: Principal Component Analysis (PCA, (Pearson, 1901)) is a dimensionality reduction technique that converts high-dimensional data into a smaller number of independent variables. We performed PCA using the t-SNE toolkit (Maaten and Hinton, 2008) for the ASAP data set, reducing it to 500 features. As a second option, we consider feature selection by frequency, following the rationale that features occurring in only a few items are less helpful: in the *frequency*-filtered feature set condition, we only use features that occur in at least 20 answers.

First, we compare whether these two feature selection methods are beneficial for unsupervised

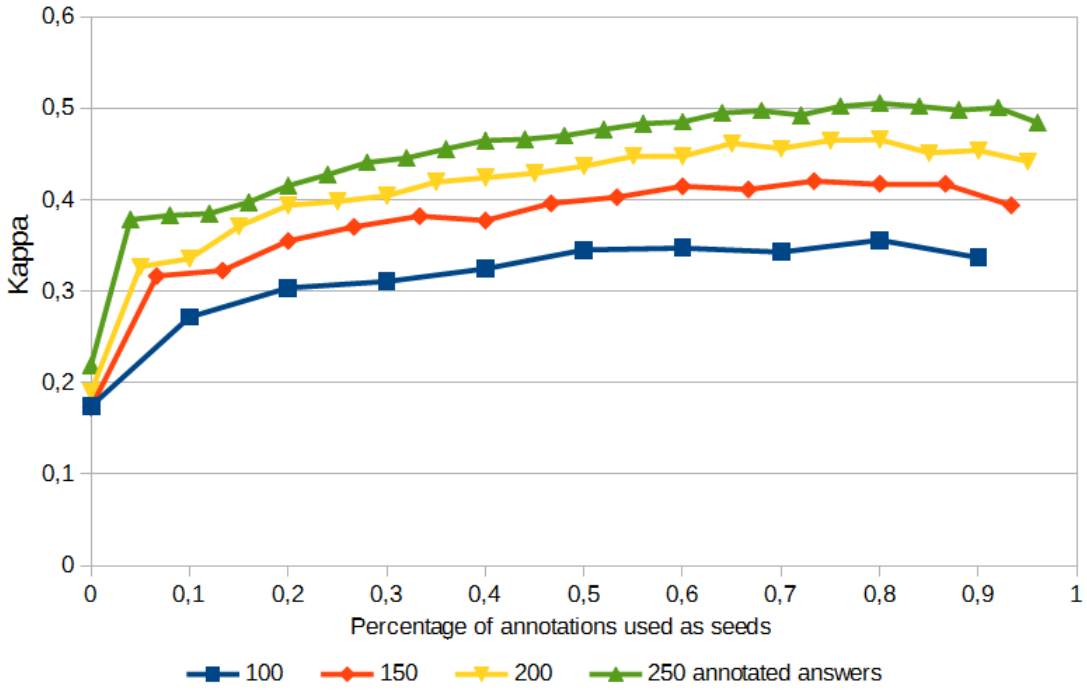

Table 6.19: Tradeoff between the number of seeds and the number of clusters for different overall amounts of human annotation steps.

features	KM_{all}	MKM	MPCKM
all features	0.504		0.566
PCA	0.323	0.317	0.374
frequency-filtered	0.500	0.489	0.563

Table 6.20: Unsupervised feature selection for two versions of completely unsupervised clustering: k-means (KM) and k-means with metric learning (MKM) and unsupervised feature selection as a preprocessing step for semi-supervised clustering (MPCKM).

k-means clustering, i.e., we compare to the unsupervised baseline KM_{all} with 150 cluster from Experiment 1 that uses all features. To account for the fact that metric learning as used in the MPCKM algorithm might be beneficial even in the absence of constraints, we also evaluate using metric learning without pairwise constraints (MKM). Results are presented in the second and third column of Table 6.20.

We see that neither of the unsupervised feature selection methods helps for KM_{all}. We also see that metric learning, which was beneficial in combination with constraints in Experiment 1, does not help here: MKM performance decreases compared to KM_{all}. PCA shows in general a much worse performance on both the KM_{all} and MKM conditions.

Next we explore whether a combination of unsupervised feature reduction with *semi-supervised clustering* helps. To do that we run the MPCKM algorithm from Experiment 1 including supervised feature selection, but with each of the two unsupervised feature reduction methods as a preprocessing step. We can see in the last column of Table 6.20 that we do not beat our previous results using PCA, but reach very similar results with frequency-reduced features. Such unsupervised feature selection methods thus provide the additional benefit of reducing runtime substantially, and we will investigate them in future work.

Experiment 4: Reusing Seeds for Label Propagation

In this experiment, we examine how seeds selected before clustering can be reused for labeling cluster centroids after clustering. In Experiments 1 to 3, we have selected the seeds for feature selection and constraints randomly; and by chance some seeds will overlap with cluster centroids, which have to be labeled for label propagation. Our goal in this experiment is to select seeds in such a way that they will have a higher overlap with the cluster centroids. For random seed selection and our setting with 40 clusters and 110 seeds out of 1000 answers, we can expect to find on average 11% of the cluster centroids among the seeds, i.e., on average 4.4 out of 40

centroids.

We increase this random overlap through an informed iterative selection of seeds: we start with a small set of 20 initial seed items for clustering. We then select one new seed at a time based on the previous clustering, re-cluster and repeat this procedure until 110 seeds (as in experiment 1) are reached. We use a sampling strategy inspired by *diversity sampling* in AL, cf. (Brinker, 2003) where the goal is to cover the complete feature space. In order to cover as many clusters as possible by our seeds, we select the cluster with the lowest frequency of labeled items (and the bigger one in case of ties) to choose the new seed for the next iteration. In order to get a good representative for that cluster that will be reusable in label propagation, we choose the item closest to the centroid as the next one to be labeled. To avoid artifacts of randomization, we average all results over 5 random seed sets per prompt.

We find that selecting seeds through *diversity sampling* increases the overlap between seeds and cluster centroids to on average 11; the actual clustering performance does not differ substantially from random sampling. Those saved 12 human annotation steps can of course be used as additional seeds in our assumed setup of 150 available human annotation steps. We thus use an additional annotation setup, where we keep adding seeds using diversity sampling until the total number of labeling steps reaches or surpasses a fixed number of labeling steps for the first time. (As the number of actually labeled data does not always increase completely linearly in each sampling step, we adapted this value to 148 instead of 150 in order to make sure that we do not label on average more than 150 items.) In that setup, we get some further performance improvement up to on average **0.577**, our overall best result for 150 human annotation steps.[4]

6.3.5 Conclusions

In this study, we have examined semi-supervised clustering methods for short answer scoring in a scenario where a set of items has to be graded with a fixed limited amount of human annotations. We have shown how this limited effort can best be used in the form of seeds for feature selection and constraints and post clustering for centroid-based label propagation. We have found that using MPCKM clustering with pairwise CL constraints and metric learning combined with supervised feature selection brings a large performance boost that (i) cannot be reached using unsupervised methods alone and (ii) comes closer to the performance of supervised machine learning methods. Selecting seeds based on diversity additionally reduces human effort as such seeds can be efficiently used for label propagation without having to label new examples.

[4] We also investigated a more extreme reusing strategy that reuses any seed contained in a cluster for label propagation (not just the centroid), but found it to be less effective, potentially, because less prototypical items are used for label propagation.

One direction for future work would be to also explore the usage of different similarity metrics such as sentence similarity of answer pairs, which are potentially highly useful for clustering, but not applicable in an ML-based approach.

6.4 Conclusions

This section presented three studies on assisted short-answer scoring where a reduction of human scoring effort is always achieved by labeling only part of data. These data can be selected either by means of active learning or by labeling only selected items in a clustering setup with label propagation.

We found that active learning methods lead to a better performance compared to a random item selection using the same number of items. Especially uncertainty-based methods are beneficial, and it helps to have all classes equally represented in the initial seed set of labeled data.

When using clustering methods, we found that especially for short answers, clusters can be formed based on surface features such as word and character n-grams. For longer answers, clustering is less effective. We found that we can use label propagation methods to assign a label: the teacher only labels a single instance, ideally the one closest to the cluster centroid. The label of the centroid is then propagated to all members in the cluster. We can improve the performance of such a clustering based scoring setup, if we use the human scoring effort not only for labeling cluster centroid, but also to label some seed items before clustering that are used to select features for the clustering process and as constraints in semi-supervised clustering. If we compare clustering to supervised machine learning, we find that we often cannot beat the supervised machine learning approach However, clustering offers an additional advantage to provide the teacher with groups of similar answers, so that she may get a quick overview of common misconceptions and give feedback to groups of students.

In a real-life situation, determining an exact number of items to be labeled will always be a matter of finding a good trade-off between grading accuracy and human scoring time investment. This trade-off depends both on the available human resources, as well as the nature of the test and how costly mis-classifications are. Whether a student might get a few points too much or too little in a homework assignment, whether they end up in a wrong course level after a placement test or whether they will fail an exam because of some misplaced answers are examples of different requirements on reliability of the scoring that require different time investments from a teacher.

Assisted scoring is thus always a trade-off between classification performance and human intervention effort, and the specific trade-off between scoring time and scoring performance is hard to predict. In the future, it will therefore be interesting to see assisted scoring scenarios

brought to real-life and see how teachers interact with scoring machineries supporting assisted scoring.

7 Processing of Learner and Student Language

In this chapter, we investigate the influence of properties of learner language on NLP processing tasks. This language variant deviates substantially from the language of proficient writers, both in terms of the vocabulary used and because of spelling and grammar variations. As an example, consider the following learner answer from the CREG corpus with all deviations from the standard marked in red. We see grammatical errors, such as the wrong gender in *"seiner"* instead of *"ihrer"*, a spelling mistake (*"durfen"* instead of *"dürfen"*), an erroneous lower-case spelling of the noun *"Wein"* and a wrong lexical choice for the verb of the second sentence, where *"nennen"* would be more appropriate than *"heißen"*.

(7.1) **Question:** Welche Konsequenzen hat das neue, von der EU geplante Gesetz für Hessen? Nennen Sie zwei bestimmte Konsequenzen.
Learner answer: Sie hätten eines seiner bedeutendsten Identifikationssymbole verloren. Sie durfen Apfelwein und Kirschwein nicht mehr "wein" heissen.

Whenever we want to automatically deal with language, some sort of linguistic processing is used. Tools such as POS-Taggers are a part of many NLP pipelines used to process learner data. In the automatic scoring approaches in Chapter 5, POS tagging is for example used in order to make sure that only words from the same POS class can be aligned. However, the performance of such preprocessing tools, which are typically trained on newspaper text, decreases substantially if they are applied to texts with other kinds of language, such as transcribed spoken language, data from computer-mediated communication or, as in our case, learner language. In the case of learner language, this is due both to domain differences, i.e., a different vocabulary, as well as spelling mistakes and ungrammaticalities in the language input. Therefore either an adaptation of tools towards non-standard language or a normalization of non-standard language, or both, is desirable for better preprocessing performance.

In this study, we consider the task of POS tagging, as it is a core part of many NLP toolchains and provides input for higher-level processing steps such as algorithms for scoring the content of learner answers. For the German CREG corpus (see Section 3.2 for an overview), we perform

a gold-standard normalization on token level and compare the performance of processing tools on the original version of the data to that of the normalized version.

Goals of this study

This section addresses research question 3.1

RQ 3.1: How can we improve linguistic preprocessing of learner answer data?

In more detail, we address the following questions:

- How reliably can normalization information be annotated for learner language, and how easy is it to annotate learner language with POS information?
- How can we exploit the reading comprehension task structure to automatically normalize learner and student data?
- How does the (gold-standard) normalization of student and learner data influence the performance of POS tagging?

Contributions

This chapter makes the following contributions: First, we describe a method to improve the POS tagging performance on learner language from German reading comprehension exercises by automatically inducing POS information for out-of-vocabulary (OOV) tokens. Second, we provide POS and normalization annotation on top of the CREG corpus (Meurers et al., 2011c). This resource is then used to evaluate our proposed method. Finally, we investigate the influence of normalization for CREG on the task of POS tagging.

7.1 The Challenges of Processing Learner Language

We use the term *learner language* to refer to written or spoken language produced by non-native speakers. The term is closely related to that of *interlanguage*, a term coined by Selinker (1972) that describes the language system of a language learner, whose utterances differ from those that a native speaker of the language would use to express the same meaning. While interlanguage is used by Selinker (1972) to describe the individual language variety used by a particular language learner (different learners have different interlanguages), we use learner language here to denote in general language output produced by non-native speakers.

	CREG
Tagset	STTS
Tagger	TnT
Tagging accuracy with standard model	92.8 %
% of OOV tokens with standard model	10.5 %

Table 7.1: POS tagging performance with a standard model on the CREG data set.

The challenge when processing any type of non-standard language is that out-of-the-box POS tagging models are usually trained on *standard language* like newspaper articles, and consequently also perform best on newspaper text. Both learner and student language, however, typically differ substantially from newspaper data, so that out-of-the-box models are not directly applicable in an online assessment setting. The standard model of the TnT tagger considered in our experiments, for instance, achieves an accuracy of about 97 % when trained on and applied to disjoint sets of German newspaper text (Brants, 2000), but only 93 % when applied to learner language, as we will see later.

One important reason for this performance drop are the out-of-vocabulary (OOV) tokens, i.e., tokens that the tagger has not seen during training. In the CREG corpus 10.5 % of all tokens are OOV. Compared to about 12 % OOV tokens on standard newspaper data (Brants, 2000), this number might at first glance seem relatively low; given the restricted vocabulary size of language learners the number are indeed quite high. OOV words are relatively frequent in learner language for two reasons: One is that the vocabulary used in the CREG corpus differs from that of typical newswire texts. We call these OOV words, which are valid words, but just happened not to occur in the training data for the POS model, *lexical gaps* in the tagger lexicon. The second and more important reason is that learner language contains a lot of noise such as typos and other spelling errors, as well as grammatical errors. They lead to nonexistent word forms which we call *misspellings*. Since learner language also tends to differ on the syntactic level, the tagger often cannot exploit contextual information to guess the correct POS tag for OOV as effectively as in the case of standard language. We further assume, that most OOV words in standard language are proper nouns and other infrequent content words where the context helps to assign the correct POS tag.

Examples (1) and (2) illustrate these phenomena: The first example shows two typical types of misspelling errors at the lexical level: The noun *"Erflog"* (correct form: *"Erfolg"*, English.: *"success"*) is a spelling error with a letter swap; *"geld"* (correct form: *"Geld"*, English: *"money"*) is written in lowercase although in German all nouns are capitalized. Due to these

errors a standard tagger tags the two nouns as a verb in the former and an adjective in the latter case. In the second example, the noun *"Dusche"* (English: *"shower"*) is spelled correctly, but it is a *lexical gap*, because the word does not occur in the training data. The tagger chooses erroneously to tag the word as an adjective instead of a noun.

(1) **Learner:** Viel **Erflog** und **geld** haben
Normalized: Viel **Erfolg** und **Geld** haben
Tagger: ADV **VVFIN** KON **ADJA** VAINF
Gold: ADV **NN** KON **NN** VAINF

(2) **Learner:** Der Herd und die **Dusche**
Normalized: Der Herd und die **Dusche**
Tagger: ART NN KON ART **ADJA**
Gold: ART NN KON ART **NN**

Most domain adaptation methods from related work on the domain of computer-mediated communication like Han et al. (2012), Rehbein (2013) or Prange et al. (2015) are not applicable in our scenario, as they rely on large in-domain corpora in order to exploit commonalities of the type of language under consideration. Substantially large learner-language corpora, however, are not readily available and additionally, learner language lacks the systematicity of other non-standard texts: Whereas in, e.g., language from forum or chat data, some abbreviations, emoticons or contractions are very frequent, language learners produce errors and deviations from the standard on many levels, and there is no usable systematicity apparent in the small data set we have at hand.

Instead our approach is to address the problem by exploiting the structure of reading comprehension questions, a typical language learning task: Students' answers are linked to a reading text in standard language; when students answer reading questions, they rely on the given textual material and tend to repeat words or even copy whole phrases from the question or the reading text (referred to as *reference* texts below), a common language learning strategy known as *lifting* (Anderson and Roit, 1996). In Section 4.2 we have shown that learner answers in CREG can often reliably be linked to individual sentences from the reference texts. Further, we observe that learner answers tend to have a high lexical overlap with the reference texts. This has implications which we leverage in our tagging approach: Whenever a word in a learner answer does not occur in the vocabulary of the tagger, we check whether this word or a similarly spelled word occurs in the reference texts. If that is the case, we assume that the learner really meant this word: We are therefore able to normalize misspellings to words occurring in the reference texts. We add words identified as lexical gaps that occur in the reference to the tagger lexicon together with the POS tag that has been assigned to them by the tagger in the reference texts. As

the reference texts are standard language and therefore tagged with a high degree of accuracy (97 %, i.e., the same accuracy as standard newspaper text), we trust their POS annotation and propagate the POS label of OOV words in the reference back to the corresponding token in the learner answer.

Consider the following example of a learner answer that contains the OOV word *"verlossen"*. In the reference, the word as is does not occur; therefore we assume it to be a misspelling and not a lexical gap in the tagger lexicon. While standard spell checkers would correct the word as either a form of *"verlassen"* or one of *"verlieren (verloren)"* we find the correct variant by comparing the word to all words occurring in the reference that contains *"verloren"*, but no instance of *"verlassen"*.

(3) **Learner Answer:** Apfelwein ist einer traditionalle Wein für ein hundert Jahre. Eine Konsequenz ist Kultur **verlossen**. (...)
Text: (...) Würde dieser Begriff verboten, hätte das Land Hessen eines seiner bedeutendsten Identifikationssymbole **verloren** (...)

To evaluate the effectiveness of our approach, we manually annotate the learner answers within CREG with POS tags and normalization information, i.e., correct word forms of incorrect learner words. The added value of our annotations compared to previous annotation efforts such as the FALKO corpus (Reznicek et al., 2012) is that we provide both manual POS tags and normalization information. It is also grounded in the structure of the CREG corpus, whose reference texts allow both a context-aware manual and automatic normalization. Our automatic tagging approach gives us an improvement of 10 percentage points tagging performance on OOV words.

7.2 Background and Related Work

Learner language can differ quite substantially from standard language. Deviations from the standard occur on all levels, including, but not limited to spelling, lexicon, syntax, and morphology. They are not universal for all learners, but depend on factors such as native language, current level in the foreign language and learning strategies. Selinker (1972) coined the term *interlanguage* for these language variants of individual learners.

Those differences from standard language affect automatic POS tagging most notably on the areas of spelling variance, punctuation, morphology, and word order. However, individual differences make it hard to build one POS tagger model for all learners. Instead of building a single tagger model, we therefore make use of what we know about the context of the learning task and exploit information from the reference material to make assumptions about the target hypothesis. In doing so, we adopt a common way of coping with learner language in NLP applications:

the integration of a normalization (spelling error correction) step into a linguistic pipeline that tries to bring the input closer to standard language. (Another option is to build individual models for single learners or groups of learner with the same characteristics. We did not consider this approach due to data sparsity.)

The task of (manual as well as automatic) normalization of learner language has been addressed in several works, for German data most notably in the FALKO corpus (Reznicek et al., 2012). This corpus consists of summaries and essays written by language learners and native speakers. Similar to our annotations, the FALKO corpus provides (manual) normalization information on several linguistic levels. Their *minimal target hypothesis* has the aim of transforming the text into a parsable structure to enable automatic processing, while the *extended target hypothesis* also remedies errors on semantic, lexical, pragmatical, and stylistic levels. The corpus also comes with POS information, but unlike us they use automatically assigned POS tags on the minimal target hypothesis and do not provide a manual POS annotation.

Reznicek and Zinsmeister (2013) provide a study about automatic POS tagging performance for learner texts from a subcorpus of FALKO. They evaluate only those tokens where an ensemble of taggers disagree and manually annotate only a very small data sample of learner essays. Also, as FALKO consists primarily of essay data, our approach of using the reference texts would not be directly applicable here.

POS annotation of learner language with a completely different goal has been approached by Díaz-Negrillo et al. (2010), who annotated NOCE, a learner corpus of English texts written by Spanish learners. Their approach differs from ours in that they are not normalizing learner language into standard language, but explicitly deal with properties of learner language by annotating separately the three individual sources of evidence for a POS tag: lexicon, morphology, and distribution. This approach allows them to identify sources of errors and to query the corpus searching for particular learner language phenomena and is thus a valuable resource for both researchers and teachers in the study of learner language. With our different goal of improving POS tagging to enable automatic linguistic processing, we are instead trying to fit learner language into the framework of a standard tagset in order to enable higher NLP processing steps.

One can also see the adaptation of POS taggers to learner language as a problem of domain adaptation. Recent work on domain adaptation has focused on Computer-Mediated Communication (CMC) data. For instance, Gadde et al. (2011) leverage word clusters based on surface similarity to link OOV words from an SMS corpus to known words and, similar to us, they also use language models to find the most plausible normalized sentence variant as a preprocessing step for tagging. Han et al. (2012) create a normalization dictionary for OOV words from English Twitter data based on distributional similarity and rank them based on string similarity.

We see one main difference between our learner language corpus and other resources of non-

standard language: For the methods described above to work, OOV words have to be frequent enough in some untagged corpus that they can be covered, e.g., in distributional models. For many CMC domains such as Twitter, large untagged corpora are available and thus many OOV tokens indeed occur with a frequency that allows relation to known words to be learned from unannotated data. Such approaches do not work in our case because of the small size of our corpus. We overcome this problem by instead leveraging information from the reading material that narrows down the pool of potential replacement candidates.

7.3 Corpus Study: Normalization and POS Tagging of German Learner Data

This section describes the annotation project that adds both part-of-speech and normalization information to the learner answers in the CREG corpus.

We organize the annotation into two subsequent steps. In a first step, we normalize the input, i.e., we replace incorrect words in the learner answers with the words that the learner presumably wanted to use (the *target hypothesis*). In a second step, we then label the words in the learner answer with POS information based on the target hypothesis. The normalization step is a necessary prerequisite to POS annotation since the POS annotation should reflect what the learner intended to express (on the lexical level). We thus decided to make the target hypothesis explicit as a separate annotation layer in order to make the POS annotations as transparent as possible and also to provide a gold standard for automatic normalization approaches.

7.3.1 Data

As in our previous corpus studies in Chapters 4 and 5, we use CREG-1032, the Corpus of Reading Comprehension Exercises (Meurers et al., 2011b), as the basis of our annotations (see Section 3.2). For our annotations, we focus on the learner language material, i.e., the learner answers. We use only those answers given primarily in German and exclude the small number of English language answers. Answers had been transcribed twice in the corpus; we always use the first transcript and tokenize it using the Stanford CoreNLP tokenizer (Manning et al., 2014). We annotate a total of 12175 learner answer tokens. In our approach to automatically improve POS tagging on learner answers, we also make use of the additional material, reading texts and questions, which consists of standard language data and is lexically related to the learner language data.

7.3.2 Normalization

In cases where a learner answer deviates from standard language we asked our annotators to form a target hypothesis, i.e., to formulate what the language learner presumably intended to say. We distinguish two normalization levels: on the first level (N1), we normalize misspellings; on the second level (N2), we additionally normalize grammatical errors such as incorrect case assignments or missing articles or prepositions. Our system, described in the next section, uses information from level N1 only; level N2 is used only in the evaluation to estimate an upper bound of tagger performance.

Consider Example 4, where the learner used *"das"*, which is a definite article in standard German, but most likely wanted to use the subordinate conjunction *"dass"* (*"that"*). This mistake was potentially due to the phonological similarity of the two words. While a word with the surface form *"das"* in German can only be tagged as an article, relative or demonstrative pronoun, only the form *"dass"* can occur as a conjunction and was presumably intended in this sentence. Therefore, the annotator first normalized *"das"* to *"dass"* before tagging the word as subordinate conjunction KOUS. Additionally, we see in this example a normalization where the learner uses an untypical spelling for the German umlaut in *"für"* and mixed up accusative with dative for the personal pronouns.

(4) LA: Sie dachte **das** es war nicht **fuer ihr** .
N1: Sie dachte **dass** es war nicht **für ihr** .
N2: Sie dachte **dass** es war nicht **für sie** .

POS:	PPER	VVFIN	**KOUS**	PPER	VAFIN	PTKNEG	APPR	PPER	$.

She thought that it was not for her.

Note that it is possible that a token is normalized on both levels. For example, the word *"großere"* in the phrase *"großere Budget"* is first spell-corrected to *"größere"* on N1 and then the case is adjusted to *"größeren"* on N2.

Our perspective on normalization as a prerequisite for POS annotation motivates our main annotation guideline: Whenever possible we only normalize on token level, i.e., we do not insert or delete words and especially do not change word order, apart from the following three exceptions:

- separate one (accidentally fused) word into two individual words (*"einsman"* becomes to *"eins man"*)
- combining two words into one word, mostly for German compound nouns which have to

be written as one token, but are often split by the English native learners (*"Polizei Gewalt"* becomes to *"Polizeigewalt"*)

- adding and deleting articles and prepositions

Unlike Reznicek et al. (2012) we do not concern ourselves with lexical or pragmatic aspects in the normalization.

Annotation Process

Annotations were carried out by two trained computational linguistics students (native speakers of German). For each learner answer they had access to the reference material, so that they could form their target hypothesis based on the context of an answer. Cases of disagreement were checked by a third annotator, who adjudicated instances that were clearly annotation errors. As Lüdeling (2008) pointed out, it is often difficult, if not impossible, to find exactly one target hypothesis. Therefore, if two different target hypotheses were both plausible they were both kept as alternatives, to maintain the diversity of potential linguistic interpretations of an answer.

Analysis

On the binary task of whether an item has to be normalized or not, annotators reached an inter-annotator agreement of $\kappa = 0.78$ for normalizations in N1 and 0.68 for normalizations in N2, indicating a substantial agreement according to Landis and Koch (1977). For those tokens where the annotators agreed to normalize them, they produced the same annotation in 86.2 % of all normalizations in level N1, and 89.2 % of all normalizations in level N2.

In the adjudication step, 47 % of all disagreements were resolved into only one correct form, while for the remaining 53 % both normalizations were kept as plausible.

After adjudication, 12.1 % of all tokens (1475) had been normalized by at least one annotator and 10.0 % (1220) by both. To test the influence of normalization on automatic POS tagging, we tagged both the normalized and the original version of the data with a standard tagger. 27.3 % of all normalized tokens changed their POS tag between these two runs.

When dividing tokens into those that are in the lexicon of a standard tagger and those that are not, we see that 35 % of all OOV tokens were normalized by at least one annotator (32 % on N1 and 6 % on N2), and only 9 % of the in-vocabulary (IV) tokens. 82 tokens were normalized on both levels.

The semantic correctness of an answer has no influence on the number of normalizations; correct learner answers contain on average as many normalizations as incorrect answers. This is a plausible result, given that teachers are instructed to ignore spelling errors when scoring short-answer questions.

	IV	OOV
N1 or N2	9 % (1023)	35 % (452)
N1	3 % (369)	32 % (414)
N2	6 % (691)	6 % (83)

Table 7.2: Number of all IV and OOV that were normalized on N1 and N2 by at least one annotator

39 % of all orthographic normalizations (N1) concerned capitalization issues and 11 % German umlaut spelling. Although we allowed some operations beyond token level, they occur rarely in our annotations: Only in 7.3 % of all normalizations in N1 did a token have to be split or two tokens were merged, and on level N2 only 14 tokens (1.8 %) were deleted and 52 (6.7 %) inserted.

7.3.3 Part-of-Speech Annotation

In the second annotation step, learner answers were annotated with POS tags using the Stuttgart-Tübingen tagset (Schiller et al., 1999). We extended the standard tagset with one extra tag, *LL*, for learner language that annotators could always use when they felt that the language used was so corrupt that no other tag would fit the token.

Annotation Process

All material was annotated by the same two student annotators with access to the reference material. As described in the normalization section, we tried to stay as close as possible to the surface form of each token and correct what we assumed to be spelling mistakes. POS tags were then selected in such a way that they fit this normalized version (step 1) of a token.

Analysis

The two annotators reached an almost perfect agreement of $\kappa = 0.95$, even if they found different normalizations. All disagreements (577) were reannotated by a third annotator. A majority vote between these three annotators was used to determine the final POS tag. 17 remaining cases of disagreement were adjudicated with one of the annotators. After adjudication 58 tokens with two possible normalizations had two different POS tags. The following example shows one such instance. In this case it is unclear, whether the learner intended to write that Mr Muschler

"could not make Julchen believe [something] (...nicht weismachen)" or that he *"could not make her believe anything (...nichts weismachen)"*. The corresponding reading text states *""Und wem gehören die Silberschuhe?", fragte sie. "Die gehören mir", sagte Herr Muschler. Aber das konnte er Julchen nicht weismachen."* and it remains unclear, whether the learner understood the meaning of the verb *"weismachen"* at all in the context of a story about painting. Therefore both interpretations seem possible.

(7.2) **Question:** Warum hörte Herr Muschler mit dem Streichen auf?
Learner Answer: Herr Muschler konnter Julchen nicht weismachen.
Target Hypothesis 1: Herr/NN Muschler/NE konnte/VMFIN Julchen/NE **nicht/PTKNEG** weismachen/VVINF ./$.
Target Hypothesis 2: Herr/NN Muschler/NE konnte/VMFIN Julchen/NE **nichts/PIS** weismachen/VVINF ./$.

The tag LL occurs for 14 tokens in the adjudicated data set. It was used rarely and – as intended – only for cases where it was not possible to assign another POS tag because the language was so corrupt. The following list shows some of the occurrences.

(5) a) Es ist etwas die besser **pkte** wohnenen.
b) **Machen** sind 66,1 Prozent Frauen.
c) Die Salzburger lassen etwas der Brunnen im Winter **fröhn** zu werden.
d) Man muss deutscher Staatsbürger sein **zu** eine GmbH gegründet werden.

Table 7.3 compares the frequency of the 10 most frequent POS tags in newspaper texts, our annotated learner data and the reference texts. We can see that learner data and reference texts have a similar distribution of POS tags that differs from that of newspaper texts.

7.4 Approaches to Automatic Normalization for CREG

This section describes the architecture of our automatic tagging system. The general aim is to minimize the number of words that are OOV to the tagger, as tagging accuracy for these words is generally much lower than for IV words. We leverage two methods to achieve this goal: One is to correct *misspelled words* before tagging into their most likely correct form, while the second is to extend the tagger lexicon with OOV words that occur in the reference material of a learner text; we call them *lexical gaps*. We first describe the decision process, for which method will be applied, and then discuss both variants of dealing with OOV words in more detail.

We use the TnT tagger (Brants, 2000) in all of our experiments, and the TIGER corpus (Brants et al., 2004) is the newspaper corpus which we use to train a standard tagger model.

POS	Newspaper	LA	References
NN	22 %	22 %	16 %
ART	11 %	10 %	8 %
APPR	9 %	5 %	6 %
ADJA	6 %	3 %	3 %
NE	6 %	4 %	3 %
ADV	4 %	2 %	5 %
VVFIN	4 %	4 %	7 %
KON	3 %	3 %	2 %
VAFIN	3 %	4 %	2 %
PPER	2 %	3 %	6 %
REST	30 %	40 %	42 %

Table 7.3: The most frequent POS tags

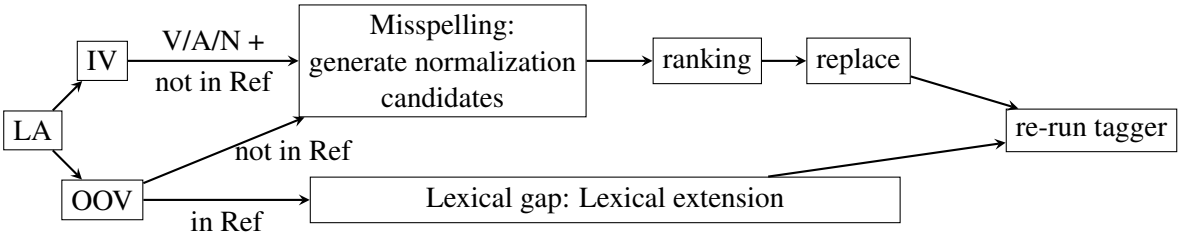

Figure 7.4: System overview with the handling of misspellings (upper branch) and lexical gaps (lower branch)

7.4.1 Decision Between Lexical Gaps and Misspellings

The lexicon of a POS tagger depends on the training corpus, i.e., all words not seen during training are OOV. Words can be OOV for two different reasons: First, it is possible that a perfectly correct word is a lexical gap and just does not occur in the training data. Second, words that are misspelled are also OOV to a tagger that has been trained on standard newspaper text.

The reference for a learner answer is a major resource for deciding whether a word belongs to the first or second of these categories. In our corpus, 84 % of all tokens in the normalized learner answers also occur in their references. This supports the claim that learners lift material from the reference into their answers. Therefore we apply the following rule: If a word that is OOV to the tagger occurs (in some inflectional form) in the reference it is likely that the learner intended to use this word (instead of misspelling a different word). We will therefore treat such a word as a lexical gap, and words that do not occur in the related material will be treated as

potential misspellings.

In the corpus, about 59 % of all OOV words indeed occur in the references and are treated as lexical gaps; the others are treated as potential spelling errors.

7.4.2 Lexicon Extension for Lexical Gaps

We determine the lexical gap words by the rule described above. Next, we determine the POS tag(s) with which words are added to the tagger lexicon. We exploit the fact that the references are mostly well-formed texts and that TnT is able to guess the correct POS tag for OOV words using suffix and context information. A sample annotation of 1000 tokens of references showed a tagging accuracy of 90 % for OOV words from the references.

Inspecting tagging results on the learner answers, we find that the accuracy on words that are lexical gaps is much lower (81 %). Our strategy therefore is to retrieve the tags that have been assigned to words in the references which are lexical gaps and include them in the lexicon. If a word occurs multiple times in the text with different POS tags, we save all of them, i.e., the word is treated as ambiguous.

7.4.3 Automatic Normalization of Misspellings

After adding entries for lexical gaps to the lexicon, we treat the remaining OOV words as potential misspellings.

Candidate Generation

We exploit the fact that most tokens in the LA stem from the reference: We compare each remaining OOV word with all words from its reference and collect all words with a Damerau-Levenshtein distance below some threshold as normalization candidates. We use a slightly modified version of the distance measure that assigns lower penalties to frequent learner issues such as capitalization and problems with German umlauts. Moreover, we compare not only to the specific word form that occurs in the reference, but extract all word forms for each lemma that is known in the TIGER corpus and compare to all of them. That means if we encounter the OOV form *"verlossen"* and have the verb infinitive *"verlieren"* in the reference, we compare *"verlossen"* also to other inflectional forms of *"verlieren"* that occur in TIGER, such as *"verloren"*.

This method only deals with OOV words, but about 25 % of orthographic corrections in our manual normalizations concern tokens whose surface form is known in the training data, for example the correction from *"das"* to *"dass"*. Therefore it is desirable to correct such misspellings that result in an IV word as well. As it might, however, introduce noise to treat every word as potentially misspelled, we restrict ourselves to IV words which do not occur in

the reference and normalize them using the TIGER corpus as reference, i.e., we take the closest word in the TIGER corpus. We only normalize words that have been tagged as content words, because function words are often very short and similar on the surface level, so that they result in a large number of orthographic neighbors.

We determine the thresholds for normalization with the reference and with TIGER individually via 10-fold cross-validation on the complete data set. See Section 7.5.4 for an evaluation of the normalization component.

Candidate Ranking

We use language models to choose between the different normalization options for a token, including the original form of the token: We combinatorically enumerate all possible normalized versions of a sentence and run them though a language model built with the SRILM toolkit (Stolcke, 2002) in order to retrieve normalizations that fit the sentence context. The language model has been trained using the Mannheimer Corpus[1] and the German part of the Wikipedia Corpus (Margaretha and Lüngen, 2014) as well as all the reading texts from the references. We keep those normalizations that constitute the variant of the sentence with the lowest perplexity.

Consider the following example sentence where two words have been automatically normalized, one with only one alternative and one with two, here listed in parentheses.

(6) Eine europäische Studie daüber (drüber, **darüber**) , worauf sie nicht vorzichten (**verzichten**) könnten

We feed all six combinatorical variants of the sentence into the language model and find the words in bold print as the ones with the highest probability and choose them as the normalization.

7.5 Experimental Study: POS Tagging for German Learner Data

In the following, we present our POS tagging experiments evaluating our proposed method.

7.5.1 Experiment 1: Baselines and Upper Bounds

As a baseline, we evaluate the out-of-the-box TnT model trained on the TIGER corpus on our annotated gold standard. As an upper bound for the normalization component, we also run the model on the normalization gold-standard version of the data. Results are reported in Table 7.5.

[1] http://www1.ids-mannheim.de/kl/projekte/korpora/archiv/mk.html

	Number of Tokens	Accuracy
Original LA	12175	92.8 %
IV	10902	95.2 %
OOV	1273	72.4 %
Normalized LA	12196 (- 12)	95.5 % (+ 2.7)
IV	11174 (+ 270)	96.6 % (+ 1.4)
OOV	989 (- 282)	82.4 % (+ 10.0)

Table 7.5: Accuracies for an out-of-the box tagging model on the original and the normalized data on OOV and in-vocabulary (*IV*) tokens.

Compared to tagging accuracy on standard texts of 96 to 97 %, the tagger performs significantly worse on our data set. Unsurprisingly, the performance is much better for the normalized LA, the accuracy gain from 92.8 % to 95.5 % bringing us back into the region of tagger performance on standard text.

The accuracy on OOV tokens increases by 10 % to 82 %, both because more precise contextual information leads to better tagging results and because normalization leads to a better performance of suffix heuristics used for OOV words. Also for IV tokens, we can observe an accuracy gain. One reason is that normalization from one IV token to another increases accuracy: For example, the conjunction *"dass"* misspelled as *"das"* is always mistagged as an article or pronoun, whereas the normalized version can be correctly tagged.

The difference in the total number of tokens between the two evaluations is due to some normalizations that split, merge or insert tokens. The number of OOV tokens is obviously reduced because many normalizations of misspellings result in IV words. Among the remaining 989 OOV tokens, 1.1 % were tokens in which the learner language was so corrupt that the annotators were not able to find a normalization.

We also evaluated the performance on the original partitioning of tokens into OOV and IV under the out-of-the-box model for comparison and see very similar results.

7.5.2 Experiment 2: Evaluating our Tagging Approach

Table 7.6 shows the performance of our full system (+Norm+Lex) compared to the TIGER baseline. We reach an accuracy improvement of 1 % for all and 9.1 % for OOV tokens. The improvement is statistically significant according to a McNemar test ($p<0.001$).

	Accuracy
TIGER	92.8 %
IV	95.2 %
OOV	72.4 %
+Norm+Lex	93.8 % (+ 1.0)*
IV	95.3 % (+ 0.1)
OOV	81.5 % (+ 9.1)*
+Norm	93.7 % (+ 0.9)*
IV	95.3 % (+ 0.1)
OOV	80.7 % (+ 8.3)*
+Lex	92.8 % (+ 0.0)
IV	95.2 % (+ 0.0)
OOV	72.6 % (+ 0.2)

Table 7.6: Accuracy of our system (+Norm+Lex), compared to the TIGER baseline, and to variants that use just one component. * denotes improvement compared to TIGER that is significant according to a McNemar test ($p<0.001$)

7.5.3 Experiment 3: Retraining the Tagger

One obvious alternative approach for adapting a tagger to a new domain is to train it on in-domain training data. Following Horbach et al. (2014b), we add two-thirds of our annotated data to the TIGER corpus (+*Gold*), retrain the tagger models and evaluate on the remaining 4000 tokens. Table 7.7, however, shows that this approach performs significantly worse than our our full system (+Norm+Lex).

7.5.4 Evaluation of Individual System Components

To assess the performance of the components of our system, we also evaluate them individually: The last two blocks of Table 7.6 show the results if we run our system with only one of the two components. We can see that the contribution of the normalization is much more pronounced than that of lexical extension, and that the combination of the two brings some additional advantage over the individual improvements.

Performance of Lexicon Extension Additionally, we evaluate the automatic generated lexicon extensions: Two human annotators determine the POS tags of the added words as a gold standard for our automatically retrieved tags. We found that about 90 % of all words are cor-

	Accuracy
TIGER	92.8 %
IV	95.2 %
OOV	72.4 %
+Gold	93.2 % (+ 0.4)*
IV	95.4 % (+ 0.2) *
OOV	74.3 % (+ 2.1) *
+Norm+Lex	93.8 % (+ 1.0) **
IV	95.3 % (+ 0.1)
OOV	81.5 % (+ 9.1)**

Table 7.7: Accuracy of a straightforward retraining approach (+Gold) compared to our system (+Norm+Lex). * denotes improvement compared to TIGER that is significant according to a McNemar test ($p<0.001$); ** denotes improvements compared to TIGER and +Gold that are significant according to a McNemar test ($p<0.001$)

rectly tagged. Frequently, errors are confusions between normal nouns and named entities.

Performance of Normalization We compare our normalization method against our gold-standard normalizations and compare the results to a baseline produced by running the spell checker GNU Aspell[2] and taking the first proposed normalization. In Table 7.8 the results are summarized: First we evaluate the binary decision on whether a token should be normalized in terms of precision, recall and F-score. Next, if a token is normalized both in the gold standard and by our normalizer or Aspell respectively, we compare how often the correct normalization is found. In both aspects our system produces better results than a state-of-the-art spell checker. The F-score increases by 20 %. If a normalization is in the right place, in 86 % of all cases it is exactly the same as in the manual annotation for our system.

Evaluation of Language Models The normalization alternatives were ranked by a language model. The language model distinguishes between an average of 2.2 alternatives. In 86 % of all cases one of them is the correct one, and in 69 % the language model ranks the correct one highest.

[2] `aspell.net`

	D-L-Distance+Ref	Aspell
Precision	0.85	0.51
Recall	0.57	0.44
F-Score	0.69	0.47
Correct Token	0.86	0.54
Correct Lemma	0.89	0.62

Table 7.8: Aspell vs. Damerau-Levenshtein Distance; upper half: number of normalizations on the right tokens; bottom half: number of all tokens with the right normalization

7.5.5 Analysis

Table 7.9 shows precision, recall, and F-score for the out-of-the-box model and the improvements by our approach. For the sake of simplicity, we merged similar tags, e.g., all verb tags, into a coarser-grained label. We can make two observations: First, the performance of the TIGER model on learner data is in general reduced; there are no prominent outliers, apart from quite infrequent classes. Second, the improvement we get from our approach manifests across all POS tags.

7.6 Conclusions

In this chapter, we addressed the task of POS tagging for learner language. We presented normalization annotation for the CREG corpus as well as POS annotation, providing a gold standard as an evaluation basis for further POS tagging and normalization approaches.

The structure of the CREG corpus, i.e., the fact that every piece of learner language is an answer to a specific question about a text, allows us to make educated guesses if a token is OOV a standard tagger, both through lexical extension and normalization. In doing so, we get a significant improvement in tagging performance from 92.8 to 93.8 % . On OOV words alone, we improve from 72.4 to 81.5 %.

	TIGER			+Norm+Lex		
POS	**Precision**	**Recall**	**F-Score**	**Precision**	**Recall**	**F-Score**
Adjective	93.7	84.5	88.8	+1.0	+7.5	+4.5
Adverb	93.4	84.0	88.4	+1.0	+0.7	+0.9
Preposition	96.4	97.4	96.9	+2.3	-0.7	+0.8
Determiner	98.9	97.9	98.4	+0.0	-0.1	-0.1
Cardinal	92.2	93.3	92.7	+2.8	-0.9	+1.0
FM	76.8	65.5	70.7	+0.0	+3.6	+2.0
Conjunction	95.0	96.6	95.8	+0.6	-0.2	+0.2
Noun	96.3	97.8	97.0	+1.5	+0.2	+0.9
Pronoun	92.5	96.5	94.4	+1.0	-0.6	+0.3
Particle	96.9	95.1	96.0	+0.4	+0.4	+0.4
TRUNC	75.0	100.0	85.7	+0.0	+0.0	+0.0
Verb	97.6	96.6	97.1	-0.6	+1.5	+0.4
XY	99.8	99.8	99.8	-0.2	+0.1	+0.0

Table 7.9: Precision, recall, and F-score percentage values for the out-of-the-box TIGER model and changes in performance for our approach

8 Conclusions

Within this thesis, we carried out research on the task of automatic short-answer scoring under two different view points: First, we showed how answers in a reading comprehension corpus link to both target answers and reading texts and how these relations can be used for automatic scoring. Second, we reduced human annotation effort in grading short-answer questions through clustering and active learning methods.

Addressing the connection between learner answers, target answers and reading texts in reading comprehension, we conducted a series of corpus annotation studies that highlight these relations.

We annotated source sentences which learners and teachers potentially used for constructing their answers and observed that most correct learner answers are backed up by the text; incorrect answers often have text support as well, but are frequently backed up by a part of the text which is not relevant to answer the question. This provides insights into how students answer reading comprehension questions. It also opens possibilities to create a new baseline scoring model, which checks whether learners looked for an answer in the right place. Such a model can then be used to point students at possible misunderstandings in reading comprehension. It does not require a explicit target answer, only highlighting of parts of the text which contain the right answer.

In addition to identifying such generally related text parts, we next qualified the relation between learner answers and target answers a well as reading texts by annotating entailment relations between them. While correct answers are often entailed by the target answer as well as the related text, incorrect answers do most of the time not stand in an entailment relation to the target answer, but there is often some partial overlap with the text. Surprisingly, in a number of cases answers are considered as correct by teachers, although they contain less detail than the target answer. These findings provide information on the scoring behavior of teachers, and also defy assumptions that short answer scoring and recognizing textual entailment are completely equivalent tasks. We find instead, that the tasks are only closely related. Gold-standard entailment information can improve the performance of automatic scoring, but cannot replace it.

We used the annotated links between learner answers and both reading texts and target answers in a statistical alignment-based scoring approach using methods from machine translation

and found that it performs comparable to a heuristic approach.

We examined two methods how human scoring workload can be reduced: active learning and clustering. In the active-learning approach, we scored particularly informative items first, i.e., items from which a classifier can learn most. We identified them using uncertainty-based sample selection. In this way, we reached a substantially higher performance with a given number of annotation steps compared to randomly selected answers. In the second research strand, we used clustering methods to group similar answers together, such that groups of answers can be scored in one scoring step. In doing so, the number of necessary labeling steps can be substantially reduced. This approach is particularly effective for low-stakes scoring scenarios such as placement tests, where the advantages of reduced grading time far outweigh the disadvantages of allowing for a small number of incorrectly-labeled answers.

We also compare clustering-based scoring to classic supervised machine learning, where human scoring annotations are used to train a classifier: our clustering approach comes close to supervised machine learning performance, whereas clusters provide the advantage of structured output, which helps in giving meaningful feedback to students. We were able to narrow the performance gap between unsupervised clustering and supervised machine learning by allocating the human annotation steps in the clustering approach efficiently in three ways: for feature selection prior to clustering, to score cluster centroids for label propagation and to form pairwise constraints in semi-supervised clustering.

An additional study investigated the automatic POS tagging of learner language. We manually annotated a German reading comprehension corpus both with spelling normalization and POS information. We found that we can improve the performance of automatic POS tagging by spell-checking the data using the reading text as additional evidence for lexical material potentially intended by a learner answer.

In sum, we learned more about the nature of short-answer questions, both in terms of how learners answer such reading comprehension questions as well as how teachers score them. We used these findings to improve automatic scoring. While previous work focused mainly on a comparison to the target answer, we also integrated the reading text into the scoring process. We further showed ways how to reduce the human scoring effort needed to manually score short-answer question through labeling only highly relevant or prototypical data points in active learning and clustering scenarios.

These theoretical findings have paved the way to more practical studies which will bring automatic scoring closer to the classroom. The most promising of these ideas will be discussed in the next chapter as directions for future work.

9 Directions for Future Work

The work in this thesis is primarily theoretical in nature: We conducted studies an automatic scoring of existing pre-scored data sets and assessed the reduction of human scoring effort in two different ways: a) in terms of the number of scoring steps needed to score a set of answers with a desired performance and b) the performance which can be reached with a given number of scoring steps.

An obvious next step will be to take these findings to the classroom and evaluate the proposed methods in a real-life setting. We will benefit in two ways from studies in this direction: First of all, we will learn about the advantages of automatic scoring for a teacher in an exam situation. This directly links to the studies in this thesis, as it exclusively focuses on summative feedback, i.e., feedback that informs teachers about the performance of a group of learners in the form of points or correctness information per learner answer. In addition, the proposed methods might be applied and further developed to be equally useful in giving formative feedback to students. That means they can serve as a basis to provide feedback about errors and misconceptions in an answer.

Regarding **summative feedback**, we need to understand better how teachers can benefit from automatic scoring methods. For the studies in this thesis, we used the number of answers to be manually labeled as a measurement for the time a teacher needs to spend correcting a set of answers. This is a reasonable approximation, although it ignores the fact that some answers may be particularly easy or particularly difficult to score. For example, borderline cases might need more attention and time from a human annotator than clearly correct or incorrect answers. Therefore, practical studies need to refine models on human scoring effort reduction. We should, for example, time the scoring of individual answers to identify the hard cases. Next, we could learn to predict how hard an answer, e.g., using the confidence of a scoring model.

For similar reasons, it will be necessary to validate how teachers interact with a scoring interface based on clustered answers. We evaluated different ways of label propagation for clustering, but can as well imagine a scenario where teachers score clusters holistically. In such a scenario, a teacher inspects a set of answers and assigns a label to the complete group of answers, and potentially singles out some answers in the cluster which should receive a different label from the rest. What such an approach means in term of labeling time in comparison to label propagation

will also be a question for future work.

A second aspect important in assisted scoring is annotator consistency, i.e., whether annotators can label a set of answers with a higher inter- and intra-annotator agreement if they see and review similar answers together. This might well be the case, as all instances of a certain type of misunderstanding will appear in a limited number of clusters and can be scored jointly. In contrast, a teacher labeling answers in random order might not be consistent on similar borderline, especially when one appears early in the scoring process and the other much later after a number of other items has been scored in between. Such an effect cannot be measured through simulation studies on already labeled data. This has to be investigated in a real-world scoring setup.

The task of giving **formative feedback** to students can also be informed by lessons learned from this thesis. Currently, there are no annotations available which actually show what good feedback to short-answer questions might look like. Good feedback to a student should in any case go beyond a simple "Your answer was incorrect", but instead provide more information either regarding the source of the error or about ways to improve the answer. Therefore, we consider highlighting potential sources of problems a good first step for feedback and a starting point to develop more explicit automatic feedback strategies.

To this end, we can use alignments between learner answers and target answers to show which part of an answer we assume to be correct and which parts substantially deviate from the expected target answer. Another application for reading comprehension questions is highlighting of text areas aligning to the target answers to give the student hints where to look for an answer, and to show her where an incorrect answer originated from. Since data sets documenting explicit meaningful feedback to students are not available (apart from very general labels such as "missing concept", etc.), collecting such feedback messages from teachers will be a valuable step towards automatic feedback generation. Encouraging teachers to give feedback to clusters of answers that represent the same misconception can serve as a starting point. Teachers can give feedback to large numbers of students without covering every answer separately.

Apart from these two application areas, a large number of more methodological topics are also important for the practical applicability of short answer scoring: these are related to model transfer and re-use of already trained models on new data or learner cohorts. The option to re-use a trained model in new settings might greatly increase its usability in real life, as it avoids the need to train a new model for each new prompt. For those of our models which are prompt-dependent, it will be interesting to see how such models can be adapted to new similar prompts, or to, for example, new data to a known prompt that comes in a different language.

In this thesis, we conducted lab experiments that successfully showed how human scoring

in the educational domain can be supported through automatic and assisted scoring methods. Overall, we think that content scoring methods in general should be taken to the classroom in order to develop them further based on real-world application scenarios in addition to theoretical considerations like those in this thesis. Only then we will be truly able to assess their benefits in teaching and testing.

Bibliography

Eneko Agirre, Mona Diab, Daniel Cer, and Aitor Gonzalez-Agirre. Semeval-2012 Task 6: A pilot on semantic textual similarity. In *Proceedings of the First Joint Conference on Lexical and Computational Semantics-Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation*, pages 385–393. Association for Computational Linguistics, 2012.

Eneko Agirre, Aitor Gonzalez-Agirre, Inigo Lopez-Gazpio, Montse Maritxalar, German Rigau, and Larraitz Uria. Semeval-2016 Task 2: Interpretable semantic textual similarity. *Proceedings of SemEval*, pages 512–524, 2016.

Salem Alelyani, Jiliang Tang, and Huan Liu. Feature selection for clustering: A review. In *Data Clustering: Algorithms and Applications*, pages 29–60. 2013. URL `http://dblp.uni-trier.de/db/books/collections/aggarwal2013.html#AlelyaniTL13`.

Enrique Amigó, Julio Gonzalo, Javier Artiles, and Felisa Verdejo. A comparison of extrinsic clustering evaluation metrics based on formal constraints. *Information Retrieval*, 12(4):461–486, August 2009. ISSN 1386-4564. doi: 10.1007/s10791-008-9066-8. URL `http://dx.doi.org/10.1007/s10791-008-9066-8`.

Enrique Amigó, Julio Gonzalo, Javier Artiles, and Felisa Verdejo. A comparison of extrinsic clustering evaluation metrics based on formal constraints. *Information retrieval*, 12(4):461–486, 2009.

Valerie Anderson and Marsha Roit. Linking reading comprehension instruction to language development for language-minority students. *The Elementary School Journal*, 96(3): 195–309, 1996. URL `http://www.jstor.org/stable/pdf/1001759.pdf?_=1470040478978`.

Stacey Bailey. *Content Assessment in Intelligent Computer-Aided Language Learning: Meaning Error Diagnosis for English as a Second Language*. PhD thesis, The Ohio State University, 2008.

Stacey Bailey and Detmar Meurers. Diagnosing meaning errors in short answers to reading comprehension questions. In *Proceedings of the Third Workshop on Innovative Use of NLP for Building Educational Applications*, pages 107–115, Columbus, Ohio, June 2008. URL http://www.aclweb.org/anthology-new/W08-0913.

Colin Bannard and Chris Callison-Burch. Paraphrasing with bilingual parallel corpora. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 597–604, 2005. URL http://www.aclweb.org/anthology/P05-1074.

Regina Barzilay and Noemie Elhadad. Sentence alignment for monolingual comparable corpora. In Michael Collins and Mark Steedman, editors, *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing*, pages 25–32, 2003. URL http://www.aclweb.org/anthology/W03-1004.

Regina Barzilay and Kathleen R. McKeown. Extracting paraphrases from a parallel corpus. In *Proceedings of 39th Annual Meeting of the Association for Computational Linguistics*, pages 50–57, Toulouse, France, July 2001. Association for Computational Linguistics. doi: 10.3115/1073012.1073020. URL http://www.aclweb.org/anthology/P01-1008.

Sumit Basu, Chuck Jacobs, and Lucy Vanderwende. Powergrading: A clustering approach to amplify human effort for short answer grading. *Transactions of the Association for Computational Linguistics*, 1:391–402, 2013. ISSN 2307-387X. URL https://tacl2013.cs.columbia.edu/ojs/index.php/tacl/article/view/139.

Luisa Bentivogli, Peter Clark, Ido Dagan, and Danilo Giampiccolo. The sixth pascal recognizing textual entailment challenge. In *Proceedings of the Third Text Analysis Conference (TAC 2010)*, 2010.

Mikhail Bilenko, Sugato Basu, and Raymond J. Mooney. Integrating constraints and metric learning in semi-supervised clustering. In *Proceedings of the Twenty-first International Conference on Machine Learning*, ICML '04, pages 81–88, New York, NY, USA, 2004. ACM. ISBN 1-58113-838-5. doi: 10.1145/1015330.1015360. URL http://doi.acm.org/10.1145/1015330.1015360.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3(Jan):993–1022, 2003.

Sabine Brants, Stefanie Dipper, Peter Eisenberg, Silvia Hansen, Esther König, Wolfgang Lezius, Christian Rohrer, George Smith, and Hans Uszkoreit. TIGER: Linguistic interpretation of a

German corpus. *Research on Language and Computation*, 2(4):597–620, 2004. ISSN 1570-7075.

Thorsten Brants. TnT – A Statistical Part-of-Speech Tagger. In *Proceedings of the Sixth Conference on Applied Natural Language Processing*, pages 224–231, Seattle, Washington, USA, April 2000. Association for Computational Linguistics. doi: 10.3115/974147.974178. URL `http://www.aclweb.org/anthology/A00-1031`.

Klaus Brinker. Incorporating diversity in active learning with support vector machines. In *IProceedings of the 20th International Conference on Machine Learning*, pages 59–66. AAAI Press, 2003.

Michael Brooks, Sumit Basu, Charles Jacobs, and Lucy Vanderwende. Divide and correct: Using clusters to grade short answers at scale. In *Proceedings of the First ACM Conference on Learning @ Scale Conference*, L@S '14, pages 89–98, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-2669-8. doi: 10.1145/2556325.2566243. URL `http://doi.acm.org/10.1145/2556325.2566243`.

Steven Burrows, Iryna Gurevych, and Benno Stein. The eras and trends of automatic short answer grading. *International Journal of Artificial Intelligence in Education*, 25(1):60–117, October 2014. ISSN 1560-4292.

Chris Callison-Burch. Syntactic constraints on paraphrases extracted from parallel corpora. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 196–205, Honolulu, Hawaii, October 2008. Association for Computational Linguistics. URL `http://www.aclweb.org/anthology/D08-1021`.

David Carver. Plans, learner strategies and self direction in language learning. *System*, 12 (2):123 – 131, 1984. ISSN 0346-251X. doi: http://dx.doi.org/10.1016/0346-251X(84)90022-8. URL `http://www.sciencedirect.com/science/article/pii/0346251X84900228`.

Jacon Cohen. Weighted kappa: nominal scale agreement with provision for scaled disagreement or partial credit. *Psychological Bulletin*, 70:213–220, 1968.

Xavier Conort. Short answer scoring: Explanation of Gxav solution. In *ASAP Short Answer Scoring Competition System Description*, 2012. URL `http://kaggle.com/asap-sas/`.

Walter Daelemans, Jakub Zavrel, Ko Sloot, and Antal Van Den Bosch. TiMBL: Tilburg Memory-Based Learner, version 6.2, Reference Guide. ILK Technical Report 09-01, 2009. URL http://ilk.kub.nl/~ilk/papers/ilk9803.ps.gz.

Ido Dagan and Oren Glickman. Probabilistic textual entailment: Generic applied modeling of language variability. In *PASCAL workshop on Learning Methods for Text Understanding and Mining*, 2004.

Ido Dagan, Oren Glickman, and Bernardo Magnini. The PASCAL Recognising Textual Entailment Challenge. In *Proceedings of the First International Conference on Machine Learning Challenges: Evaluating Predictive Uncertainty Visual Object Classification, and Recognizing Textual Entailment*, MLCW'05, pages 177–190, Berlin, Heidelberg, 2006. Springer-Verlag. ISBN 3-540-33427-0, 978-3-540-33427-9. doi: 10.1007/11736790_9. URL http://dx.doi.org/10.1007/11736790_9.

Ido Dagan, Dan Roth, Mark Sammons, and Fabio Massimo Zanzotto. *Recognizing Textual Entailment: Models and Applications*. Synthesis Lectures on Human Language Technologies. Morgan & Claypool Publishers, 2013. ISBN 9781598298345. doi: 10.2200/S00509ED1V01Y201305HLT023. URL http://dx.doi.org/10.2200/S00509ED1V01Y201305HLT023.

Phil Davies. There's no confidence in multiple-choice testing. In M. Danson, editor, *6th International CAA Conference*, Loughborough University, 4-5 July 2002. URL https://dspace.lboro.ac.uk/dspace-jspui/bitstream/2134/1875/1/davies_p1.pdf.

Richard R. Day and Jeong-Suk Park. Developing reading comprehension questions. *Reading in a Foreign Language*, 1(17):60–73, 2005.

Ana Díaz-Negrillo, Detmar Meurers, Salvador Valera, and Holger Wunsch. Towards interlanguage POS annotation for effective learner corpora in SLA and FLT. *Language Forum*, 36(1–2):139–154, 2010. ISSN 0253-9071. URL http://purl.org/dm/papers/diaz-negrillo-et-al-09.html. Special Issue on Corpus Linguistics for Teaching and Learning. In Honour of John Sinclair.

Dmitriy Dligach and Martha Palmer. Good seed makes a good crop: accelerating active learning using language modeling. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2*, pages 6–10. Association for Computational Linguistics, 2011.

Bill Dolan, Chris Quirk, and Chris Brockett. Unsupervised construction of large paraphrase corpora: Exploiting massively parallel news sources. COLING, August 2004.

Nicholas Dronen, Peter W. Foltz, and Kyle Habermehl. Effective sampling for large-scale automated writing evaluation systems. *CoRR*, abs/1412.5659, 2014. URL `http://arxiv.org/abs/1412.5659`.

Myroslava O. Dzikovska, Rodney Nielsen, Chris Brew, Claudia Leacock, Danilo Giampiccolo, Luisa Bentivogli, Peter Clark, Ido Dagan, and Hoa Trang Dang. Semeval-2013 task 7: The joint student response analysis and 8th recognizing textual entailment challenge. **SEM 2013: The First Joint Conference on Lexical and Computational Semantics*, 2013.

Eugene S. Edgington. *Randomization tests*. Marcel Dekker, Inc., New York, NY, USA, 1986. ISBN 0-824-77656-9.

Rod Ellis. *The Study of Second Language Acquisition*. Oxford University Press, 1994.

Manaal Faruqui and Sebastian Padó. Training and evaluating a German named entity recognizer with semantic generalization. In *Proceedings of KONVENS 2010*, Saarbrücken, Germany, 2010.

Rosa L. Figueroa, Qing Zeng-Treitler, Long H. Ngo, Sergey Goryachev, and Eduardo P. Wiechmann. Active learning for clinical text classification: is it better than random sampling? *Journal of the American Medical Informatics Association*, 19(5):809–816, 2012.

Apache Software Foundation. OpenNLP, 2010. URL `http://opennlp.apache.org`.

Evgeniy Gabrilovich and Shaul Markovitch. Computing semantic relatedness using Wikipedia-based explicit semantic analysis. In *IJCAI*, volume 7, pages 1606–1611, 2007.

Phani Gadde, L. Venkata Subramaniam, and Tanveer A. Faruquie. Adapting a WSJ trained part-of-speech tagger to noisy text: preliminary results. In *Proceedings of the 2011 Joint Workshop on Multilingual OCR and Analytics for Noisy Unstructured Text Data*, page 5. ACM, 2011.

David Gale and Lloyd S. Shapley. College admissions and the stability of marriage. *The American Mathematical Monthly*, 69(1):9–15, 1962. ISSN 00029890. URL `http://www.jstor.org/stable/2312726`.

Danilo Giampiccolo, Bernardo Magnini, Ido Dagan, and Bill Dolan. The Third PASCAL Recognizing Textual Entailment Challenge. In *Proceedings of the ACL-PASCAL Workshop on*

Textual Entailment and Paraphrasing, RTE '07, pages 1–9, Stroudsburg, PA, USA, 2007. Association for Computational Linguistics. URL http://dl.acm.org/citation.cfm?id=1654536.1654538.

Laurent Gillard, Patrice Bellot, and Marc El-Bèze. Question answering evaluation survey. In Nicoletta Calzolari, Khalid Choukri, Aldo Gangemi, Bente Maegaard, Joseph Mariani, Jan Odijk, and Daniel Tapias, editors, *Proceedings of the Fifth International Conference on Language Resources and Evaluation, LREC 2006, Genoa, Italy, May 22-28, 2006.*, pages 1133–1138. European Language Resources Association (ELRA), 2006. URL http://www.lrec-conf.org/proceedings/lrec2006/pdf/515_pdf.pdf.

Martin Gleize and Brigitte Grau. Limsiiles: Basic english substitution for student answer assessment at semeval 2013. In **SEM, Volume 2: Proceedings of SemEval 2013*, pages 598–602, Atlanta, Georgia, USA, June 2013. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/S13-2100.

Michael Hahn and Detmar Meurers. Evaluating the meaning of answers to reading comprehension questions: A semantics-based approach. In *Proceedings of the 7th Workshop on Innovative Use of NLP for Building Educational Applications (BEA7)*, pages 326–336, Montreal, Canada, 2012. Association for Computational Linguistics. URL http://purl.org/dm/papers/hahn-meurers-12.html.

Thomas M. Haladyna and Michael C. Rodriguez. *Developing and Validating Test Items*. ISBN 9781136961977.

Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. The WEKA Data Mining Software: An Update. *SIGKDD Explor. Newsl.*, 11(1): 10–18, November 2009. ISSN 1931-0145. doi: 10.1145/1656274.1656278. URL http://doi.acm.org/10.1145/1656274.1656278.

Birgit Hamp and Helmut Feldweg. Germanet – a lexical-semantic net for German. In *In Proceedings of ACL workshop Automatic Information Extraction and Building of Lexical Semantic Resources for NLP Applications*, pages 9–15, 1997.

Bo Han, Paul Cook, and Timothy Baldwin. Automatically constructing a normalisation dictionary for microblogs. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 421–432. Association for Computational Linguistics, 2012.

Michael Heilman and Nitin Madnani. The impact of training data on automated short answer scoring performance. *Silver Sponsor*, pages 81–85, 2015.

Derrick Higgins, Chris Brew, Michael Heilman, Ramon Ziai, Lei Chen, Aoife Cahill, Michael Flor, Nitin Madnani, Joel R. Tetreault, Daniel Blanchard, Diane Napolitano, Chong Min Lee, and John Blackmore. Is getting the right answer just about choosing the right words? the role of syntactically-informed features in short answer scoring. *Computation and Language*, 2014. URL `http://arxiv.org/abs/1403.0801`.

Eli Hinkel. Integrating the four skills: Current and historical perspectives. In R. B. Kaplan, editor, *Oxford Handbook in Applied Linguistics*, pages 110–126. 2010.

Andrea Horbach, Alexis Palmer, and Magdalena Wolska. Finding a tradeoff between accuracy and rater's workload in grading clustered short answers. In *Proceedings of the 9th Language Resources and Evaluation Conference (LREC)*, pages 588–595, Reykjavik, Iceland, 2014a.

Andrea Horbach, Diana Steffen, Stefan Thater, and Pinkal Manfred. Improving the performance of standard part-of-speech taggers for computer-mediated communication. In *Proceedings of the 12th edition of the KONVENS conference Vol. 1. - Hildesheim*, 2014b.

James Jesensky. Team JJJ technical methods paper. In *ASAP Short Answer Scoring Competition System Description*, 2012. URL `http://kaggle.com/asap-sas/`.

Manfred Krifka. For a structured meaning account of questions and answers. *Audiatur Vox Sapientia. A Festschrift for Arnim von Stechow*, pages 287–319, 2001.

Klaus Krippendorff. *Content Analysis: An Introduction to Methodology*. Sage Publications, Inc., Beverly Hills, CA.

Thomas K Landauer, Peter W Foltz, and Darrell Laham. An introduction to latent semantic analysis. *Discourse Processes*, 25(2-3):259–284, 1998.

J. Richard Landis and Gary G. Koch. The measurement of observer agreement for categorical data. *Biometrics*, 33(1):pp. 159–174, 1977. ISSN 0006341X. URL `http://www.jstor.org/stable/2529310`.

Batia Laufer and Zahava Goldstein. Testing vocabulary knowledge: Size, strength, and computer adaptiveness. *Language Learning*, 54(3):399–436, 2004.

Claudia Leacock and Martin Chodorow. C-rater: Automated scoring of short-answer questions. *Computers and the Humanities*, 37(4):389–405, 2003. URL `http://www.springerlink.com/content/t1t084678ptj5084/fulltext.pdf`.

Omer Levy, Torsten Zesch, Ido Dagan, and Iryna Gurevych. Recognizing partial textual entailment. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, pages 451–455, Sofia, Bulgaria, August 4-9 2013. URL http://www.aclweb.org/old_anthology/P/P13/P13-2.pdf#page=499.

David D. Lewis and William A. Gale. A sequential algorithm for training text classifiers. In *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '94, pages 3–12, New York, NY, USA, 1994. Springer-Verlag New York, Inc. ISBN 0-387-19889-X. URL http://dl.acm.org/citation.cfm?id=188490.188495.

Stuart P. Lloyd. Least squares quantization in PCM. *IEEE Transactions on Information Theory*, 28:129–137, 1982.

Anke Lüdeling. Mehrdeutigkeiten und Kategorisierung: Probleme bei der Annotation von Lernerkorpora. *Fortgeschrittene Lernervarietäten*, pages 119–140, 2008.

Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605, 2008.

Bill MacCartney and Christopher D. Manning. An extended model of natural logic. In *Proceedings of the Eighth International Conference on Computational Semantics*, IWCS-8 '09, pages 140–156, Stroudsburg, PA, USA, 2009. Association for Computational Linguistics. ISBN 978-90-74029-34-6. URL http://dl.acm.org/citation.cfm?id=1693756.1693772.

Bernardo Magnini, Simone Romagnoli, Alessandro Vallin, Jesús Herrera, Anselmo Penas, Víctor Peinado, Felisa Verdejo, and Maarten de Rijke. The multiple language question answering track at CLEF 2003. In *Workshop of the Cross-Language Evaluation Forum for European Languages*, pages 471–486. Springer, 2003.

Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60, 2014. URL http://www.aclweb.org/anthology/P/P14/P14-5010.

Eliza Margaretha and Harald Lüngen. Building linguistic corpora from Wikipedia articles and discussions. *JLCL*, 29(2):59–82, 2014. URL http://www.jlcl.org/2014_Heft2/3MargarethaLuengen.pdf.

Andrew McCallum and Kamal Nigam. Employing EM and Pool-Based Active Learning for Text Classification. In *Proceedings of the Fifteenth International Conference on Machine Learning*, ICML '98, pages 350–358, San Francisco, CA, USA, 1998. Morgan Kaufmann Publishers Inc. ISBN 1-55860-556-8. URL `http://dl.acm.org/citation.cfm?id=645527.757765`.

Andrew Mellor. Essay length, lexical diversity and automatic essay scoring. In *Memoirs of the Osaka Institute of Technology*, number 2 in series B, pages 1–14, 2011.

Prem Melville and Raymond J. Mooney. Diverse ensembles for active learning. In *Proceedings of 21st International Conference on Machine Learning (ICML-2004)*, pages 584–591, Banff, Canada, July 2004.

Detmar Meurers, Ramon Ziai, Niels Ott, and Stacey Bailey. Integrating parallel analysis modules to evaluate the meaning of answers to reading comprehension questions. *Special Issue on Free-text Automatic Evaluation. International Journal of Continuing Engineering Education and Life-Long Learning (IJCEELL)*, 21(4):355–369, 2011a. URL `http://purl.org/dm/papers/meurers-ziai-ott-bailey-11.html`.

Detmar Meurers, Ramon Ziai, Niels Ott, and Janina Kopp. Corpus of reading comprehension exercises in German, 2011b.

Detmar Meurers, Ramon Ziai, Niels Ott, and Janina Kopp. Evaluating answers to reading comprehension questions in context: Results for german and the role of information structure. In *Proceedings of the TextInfer 2011 Workshop on Textual Entailment*, pages 1–9, Edinburgh, Scottland, UK, 2011c. Association for Computational Linguistics. URL `http://www.aclweb.org/anthology/W11-2401`.

Shachar Mirkin, Ido Dagan, and Sebastian Padó. Assessing the role of discourse references in entailment inference. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 2010.

Michael Mohler and Rada Mihalcea. Text-to-text semantic similarity for automatic short answer grading. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, EACL '09, pages 567–575, Stroudsburg, PA, USA, 2009. Association for Computational Linguistics. URL `http://portal.acm.org/citation.cfm?id=1609067.1609130`.

Michael Mohler, Razvan C. Bunescu, and Rada Mihalcea. Learning to grade short answer questions using semantic similarity measures and dependency graph alignments. In Dekang

Lin, Yuji Matsumoto, and Rada Mihalcea, editors, *ACL*, pages 752–762, 2011. ISBN 978-1-932432-87-9.

Dragos Stefan Munteanu and Daniel Marcu. Extracting parallel sub-sentential fragments from non-parallel corpora. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, ACL-44, pages 81–88, Stroudsburg, PA, USA, 2006. Association for Computational Linguistics. doi: 10.3115/1220175.1220186. URL http://dx.doi.org/10.3115/1220175.1220186.

Rodney D. Nielsen, Wayne Ward, and James H. Martin. Recognizing entailment in intelligent tutoring systems. *Natural Language Engineering*, 15:479–501, 2009.

Nobal Bikram Niraula and Vasile Rus. Judging the quality of automatically generated gap-fill question using active learning. In *Proceedings of the Tenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 196–206, Denver, Colorado, June 2015. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/W15-0623.

Franz Josef Och and Hermann Ney. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51, March 2003. ISSN 0891-2017. doi: 10.1162/089120103321337421. URL http://dx.doi.org/10.1162/089120103321337421.

Simon Ostermann, Nikolina Koleva, Alexis Palmer, and Andrea Horbach. CSGS: Adapting a short answer scoring system for multiple-choice reading comprehension exercises. In *Working notes for QA Track – CLEF Question Answering Track: Entrance Exams*, 2014.

Simon Ostermann, Andrea Horbach, and Manfred Pinkal. Annotating entailment relations for shortanswer questions. In *NLP-TEA*, 2015.

Niels Ott, Ramon Ziai, and Detmar Meurers. Creation and analysis of a reading comprehension exercise corpus: Towards evaluating meaning in context. In Thomas Schmidt and Kai Wörner, editors, *Multilingual Corpora and Multilingual Corpus Analysis*, Hamburg Studies in Multilingualism (HSM), pages 47–69. Benjamins, Amsterdam, 2012. URL http://purl.org/dm/papers/ott-ziai-meurers-12.html.

Anselmo Peñas and Alvaro Rodrigo. A simple measure to assess non-response. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, pages 1415–1424, Stroudsburg, PA,

USA, 2011. Association for Computational Linguistics. ISBN 978-1-932432-87-9. URL http://dl.acm.org/citation.cfm?id=2002472.2002646.

Karl Pearson. On the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that can be reasonably supposed to have arisen from random sampling. *Philosophical Magazine*, 50:157–175, 1900.

Karl Pearson. On lines and planes of closest fit to systems of points in space. *Philosophical Magazine*, 2(6):559–572, 1901.

Anselmo Peñas, Yusuke Miyao, Álvaro Rodrigo, Eduard H Hovy, and Noriko Kando. Overview of CLEF QA Entrance Exams Task 2014. In *CLEF (Working Notes)*, pages 1194–1200, 2014.

Jonathan Peters and Pawel Jankiewicz. The William and Flora Hewlett Foundation Automated Student Assessment Prize (ASAP). In *ASAP Short Answer Scoring Competition System Description*, 2012. URL http://kaggle.com/asap-sas/.

J. Platt. Fast training of support vector machines using sequential minimal optimization. In B. Schoelkopf, C. Burges, and A. Smola, editors, *Advances in Kernel Methods - Support Vector Learning*. MIT Press, 1998. URL http://research.microsoft.com/\~jplatt/smo.html.

Jakob Prange, Stefan Thater, and Andrea Horbach. Unsupervised induction of part-of-speech information for OOV words in German Internet forum posts. In *Proceedings of the 2nd Workshop on Natural Language Processing for Computer-Mediated Communication / Social Media (NLP4CMC)*, 2015.

Stephen G. Pulman and Jana Z. Sukkarieh. Automatic short answer marking. In *Proceedings of the second workshop on Building Educational Applications Using NLP*, EdAppsNLP 05, pages 9–16, 2005. URL http://dl.acm.org/citation.cfm?id=1609829.1609831.

Chris Quirk, Chris Brockett, and William Dolan. Monolingual machine translation for paraphrase generation. In Dekang Lin and Dekai Wu, editors, *Proceedings of EMNLP 2004*, pages 142–149, Barcelona, Spain, July 2004. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/W04-3219.

Lakshmi Ramachandran, Jian Cheng, and Peter Foltz. Identifying patterns for short answer scoring using graph-based lexico-semantic text matching. In *Proceedings of the Tenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 97–106, 2015.

Michaela Regneri and Rui Wang. Using discourse information for paraphrase extraction. In *Proceedings of EMNLP-CoNNL 2012*, Jeju, Korea, 2012.

Ines Rehbein. Fine-grained POS tagging of German tweets. In *Language Processing and Knowledge in the Web*, pages 162–175. Springer, 2013.

Marc Reznicek and Hagen Hirschmann. Competing target hypotheses in the Falko corpus. *Automatic treatment and analysis of learner corpus data*, 59:101–123, 2013.

Marc Reznicek and Heike Zinsmeister. STTS-Konfusionsklassen beim Tagging von Fremdsprachlernertexten. *JLCL*, 28(1):63–83, 2013.

Marc Reznicek, Anke Lüdeling, Cedric Krummes, Franziska Schwantuschke, Maik Walter, Karin Schmidt, Hagen Hirschmann, and Torsten Andreas. Das falko-handbuch korpusaufbau und annotationen version 2.01, 2012.

Anne Schiller, Simone Teufel, Christine Stöckert, and Christine Thielen. Guidelines für das Tagging deutscher Textcorpora mit STTS. Technical report, Universität Stuttgart, Universität Tübingen, 1999.

Helmut Schmid. Probabilistic part-of-speech tagging using decision trees. In *Proceedings of the International Conference on New Methods in Language Processing*, Manchester, United Kingdom, 1994.

Helmut Schmid. Improvements in part-of-speech tagging with an application to German. In *Proceedings of the ACL SIGDAT-Workshop*, pages 47–50, 1995.

Norbert Schmitt. *Researching vocabulary: A vocabulary research manual*. Springer, 2010.

Michael Scriven. The methodology of evaluation. In R. Tyler, R. Gagné, and M. Scriven, editors, *Perspectives of Curriculum Evaluation, AERA Monograph Series on Curriculum Evaluation*, volume 1, pages 39–83. Rand McNally, Chicago, 1967.

Larry Selinker. Interlanguage. *International Review of Applied Linguistics in Language Teaching*, 10(1–4):209–232, 1972. URL `http://www.degruyter.com/downloadpdf/j/iral.1972.10.issue-1-4/iral.1972.10.1-4.209/iral.1972.10.1-4.209.xml`.

Rico Sennrich, Gerold Schneider, Martin Volk, and Martin Warin. A new hybrid dependency parser for German. In Christian Chiarcos, Richard Eckart de Castilho, and Manfred Stede, editors, *Von der Form zur Bedeutung: Texte automatisch verarbeiten? From Form to Meaning:*

Processing Texts Automatically. Proceedings of the Biennial GSCL Conference 2009, pages 115–124. Narr, Tübingen, 2009. URL http://dx.doi.org/10.5167/uzh-25506.

Burr Settles. Active learning literature survey. *University of Wisconsin, Madison*, 52(55-66):11, 2010.

Burr Settles. *Active Learning*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan & Claypool, 2012.

Michael Steinbach, George Karypis, Vipin Kumar, et al. A comparison of document clustering techniques. In *KDD workshop on text mining*, volume 400, pages 525–526. Boston, 2000.

Asher Stern and Ido Dagan. Biutee: A modular open-source system for recognizing textual entailment. In *Proceedings of the ACL 2012 System Demonstrations*, ACL '12, pages 73–78, Stroudsburg, PA, USA, 2012. Association for Computational Linguistics. URL http://dl.acm.org/citation.cfm?id=2390470.2390483.

Asher Stern and Ido Dagan. The BIUTEE research platform for transformation-based textual entailment recognition. *Linguistics Issues in Language Technology LiLT*, 9, 2013.

Andreas Stolcke. SRILM - An Extensible Language Modeling Toolkit. In *Seventh international conference on spoken language processing*, pages 901–904, 2002.

Jana Z. Sukkarieh and Stephen G. Pulman. Information extraction and machine learning: Auto-marking short free text responses to science questions. In Chee-Kit Looi, Gordon I. McCalla, Bert Bredeweg, and Joost Breuker, editors, *Artificial Intelligence in Education - Supporting Learning through Intelligent and Socially Informed Technology, Proceedings of the 12th International Conference on Artificial Intelligence in Education, AIED 2005, July 18-22, 2005, Amsterdam, The Netherlands*, volume 125 of *Frontiers in Artificial Intelligence and Applications*, pages 629–637, 2005. ISBN 978-1-58603-530-3.

Jana Z. Sukkarieh, Stephen G. Pulman, and Nicholas Raikes. Auto-marking: using computational linguistics to score short, free text responses. In *Proceedings of the 29th annual conference of the International Association for Educational Assessment (IAEA)*, 2003.

Jana Z. Sukkarieh, Stephen G. Pulman, and Nicholas Raikes. Auto-marking 2: An update on the UCLES-Oxford University research into using computational linguistics to score short, free text responses. In *Proceedings of the Thirtieth Annual Conference of the International Association for Educational Assessment*, 2004.

Jana Zuheir Sukkarieh and John Blackmore. C-rater: Automatic content scoring for short constructed responses. In *Proceedings of the 22nd International Conference of the Florida Artificial Intelligence Research Society Conference (FLAIRS)*, 2009.

Luis Tandalla. Scoring short answer essays. In *ASAP Short Answer Scoring Competition System Description*, 2012. URL `http://kaggle.com/asap-sas/`.

Mildred C Templin. Certain language skills in children; their development and interrelationships. 1957.

Katrin Tomanek and Udo Hahn. Reducing class imbalance during active learning for named entity annotation. In *Proceedings of the Fifth International Conference on Knowledge Capture (K-Cap 2009)*, pages 105–112. ACM, 2009.

Simon Tong and Daphne Koller. Support vector machine active learning with applications to text classification. *Journal of machine learning research*, 2:45–66, March 2002. ISSN 1532-4435. doi: 10.1162/153244302760185243. URL `http://dx.doi.org/10.1162/153244302760185243`.

Arnim von Stechow. *Discourse Particles*, chapter Focusing and Background Operators. 1990.

Arnim von Stechow and Thomas Ede Zimmermann. Term answers and contextual change. *Linguistics*, 22:3–40, 1984.

Ellen M. Voorhees. The TREC question answering track. *Natural Language Engineering*, 7: 361–378, 12 2001. ISSN 1469-8110. doi: 10.1017/S1351324901002789. URL `http://journals.cambridge.org/article_S1351324901002789`.

Ellen M. Voorhees and Donna K. Harman. *TREC: Experiment and Evaluation in Information Retrieval (Digital Libraries and Electronic Publishing)*. The MIT Press, 2005. ISBN 0262220733.

Rui Wang and Chris Callison-Burch. Paraphrase fragment extraction from monolingual comparable corpora. In *Proceedings of the 4th Workshop on Building and Comparable Corpora: Comparable Corpora and the Web*, pages 52–60, Portland, Oregon, June 2011. Association for Computational Linguistics. URL `http://www.aclweb.org/anthology/W11-1208`.

Seid Muhie Yimam, Iryna Gurevych, Richard Eckart de Castilho, and Chris Biemann. Webanno: A flexible,web-based and visually supported system for distributed annotations. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (System*

Demonstrations) (ACL 2013), pages 1–6, Stroudsburg, PA, USA, August 2013. Association for Computational Linguistics.

Jure Zbontar. Short answer scoring by stacking. In *ASAP Short Answer Scoring Competition System Description*, 2012. URL `http://kaggle.com/asap-sas/`.

Torsten Zesch, Michael Heilman, and Aoife Cahill. Reducing annotation efforts in supervised short answer scoring. In *Proceedings of the Building Educational Applications Workshop at NAACL*, pages 124–132, 2015. URL `https://drive.google.com/file/d/0B-veRN2Jq4PmTjA1WldvSkxCZlU/view?usp=sharing`.

Shiqi Zhao, Haifeng Wang, Ting Liu, and Sheng Li. Pivot approach for extracting paraphrase patterns from bilingual corpora. In *Proceedings of ACL-08: HLT*, pages 780–788, Columbus, Ohio, June 2008. Association for Computational Linguistics. URL `http://www.aclweb.org/anthology/P08-1089`.

Ramon Ziai, Niels Ott, and Detmar Meurers. Short Answer Assessment: Establishing Links Between Research Strands. In *Proceedings of the 7th Workshop on Innovative Use of NLP for Building Educational Applications (BEA7)*, pages 190–200, Montreal, Canada, 2012. Association for Computational Linguistics. URL `http://purl.org/dm/papers/ziai-ott-meurers-12.html`.