Zürcher Hochschule
für Angewandte Wissenschaften

School of
Engineering

InIT Institut für angewandte
Informationstechnologie

Projektarbeit (Informatik)

Entwicklung eines Meta-Klassifikators zur Sentiment Analyse

Autoren	Lukas Aschwanden Frederic Saladin
Hauptbetreuung	Mark Cieliebak Oliver Dürr
Nebenbetreuung	Fatih Uzdilli

Datum	19.12.2014

Zürcher Hochschule
für Angewandte Wissenschaften

School of
Engineering

Erklärung betreffend das selbständige Verfassen einer Projektarbeit an der School of Engineering

Mit der Abgabe dieser Projektarbeit versichert der/die Studierende, dass er/sie die Arbeit selbständig und ohne fremde Hilfe verfasst hat. (Bei Gruppenarbeiten gelten die Leistungen der übrigen Gruppenmitglieder nicht als fremde Hilfe.)

Der/die unterzeichnende Studierende erklärt, dass alle zitierten Quellen (auch Internetseiten) im Text oder Anhang korrekt nachgewiesen sind, d.h. dass die Projektarbeit keine Plagiate enthält, also keine Teile, die teilweise oder vollständig aus einem fremden Text oder einer fremden Arbeit unter Vorgabe der eigenen Urheberschaft bzw. ohne Quellenangabe übernommen worden sind.

Bei Verfehlungen aller Art treten die Paragraphen 39 und 40 (Unredlichkeit und Verfahren bei Unredlichkeit) der ZHAW Prüfungsordnung sowie die Bestimmungen der Disziplinarmassnahmen der Hochschulordnung in Kraft.

Ort, Datum:

……

Unterschriften:

……

……..

……

Das Original dieses Formulars ist bei der ZHAW-Version aller abgegebenen Projektarbeiten zu Beginn der Dokumentation nach dem Titelblatt mit Original-Unterschriften und -Datum (keine Kopie) einzufügen.

Zusammenfassung

Bei der Sentiment-Analyse von kurzen Texten mittels Meta-Klassifikatoren, d.h. der Bündelung von mehreren Klassifikatoren zu einem, besteht noch Forschungsbedarf. Trotz Studien, die in einzelnen Fällen zeigen, dass das Konzept Meta-Klassifikator grundsätzlich funktioniert, besteht sowohl bei der Erkennungsrate als auch bei der Effizienz Optimierungsbedarf.

In der vorliegenden Arbeit wird zum einen versucht einen Meta-Klassifikator zu entwickeln und zu untersuchen, ob dieser mit steigender Anzahl beteiligter Klassifikatoren zunehmend bessere Ergebnisse erzielt. Zum anderen wird versucht ein Auswahlverfahren zu ermitteln, um bei gleichbleibender Erkennungsrate die Anzahl beteiligter Klassifikatoren zu minimieren. Als Datengrundlage werden die Sentiment-Vorhersagen von 50 Klassifikatoren zu 1853 Tweets sowie ihre zugehörigen von Menschen bestimmten Sentiments verwendet.

Zur Klärung der Fragen wird eine zuvor in Software entwickelte Versuchsanordnung benutzt, deren Kern aus einem Random-Forest-Algorithmus als Meta-Klassifikator besteht. Zur Verifikation der Versuchsanordnung wurde ein existierender Versuch im Bereich Meta-Klassifikatoren nachgebildet.

Die Versuche zeigen, dass der Meta-Klassifikator funktioniert, d.h. bessere Erkennungsraten erzielt als die einzelnen Klassifikatoren. Die Verbesserung umfasst bis zu 3.69 F1-Score-Punkte. Ausserdem unterstützen sie die These, wonach mit steigender Anzahl Klassifikatoren die Erkennungsrate des Meta-Klassifikators weiter steigt. Für das Auswahlverfahren zur Minimierung der Anzahl beteiligter Klassifikatoren konnte abgesehen von ersten Ansätzen keine stichhaltige Lösung gefunden werden.

Eine weiterführende Untersuchung mittels der bestehenden Versuchsanlage wäre von Interesse. Die nächste zu klärende Frage wäre etwa, ob möglichst unterschiedliche Klassifikatoren in einem Meta-Klassifikator zu besseren Resultaten führen.

zh
aw

Abstract

In the field of sentiment analysis of short texts with the help of meta classifiers, i.e. the bundling of multiple classifiers into a single one, there is still a need for research. Although there are studies that demonstrate the concept of meta classifiers to be valid, there is still room for improvement regarding the rate of correct predictions and efficiency.

On the one hand this paper pursuits the attempt to implement a meta classifier and research whether it is possible to achieve continuously superior results while increasing the number of involved classifiers. On the other hand it attempts to determine a selection procedure to retain a certain hit rate while minimizing the number of involved classifiers. The data basis consists of the sentiment predictions from 50 classifiers for 1853 tweets as well as the corresponding sentiments determined by human beings.

To clarify the questions, a software based experimental set-up developed beforehand is utilized, whose core comprises a random forest algorithm as meta classifier. The validity of the experimental set-up is also tested by recreating an existing experiment in the field of meta classifiers.

The experiments demonstrate that the meta classifier is functional, i.e. it is capable of achieving better results than any of the individual classifiers. The improvements are up to 3.69 F1 score points. In addition the experiments support the thesis that the hit rate of the meta classifier is continuously increasing with the number of involved classifiers. Aside from slight silver linings, it was not possible to discover a solid solution regarding the selection procedure in order to minimize the number of involved classifiers.

Ongoing research with the experimental set-up should be of interest. The next question may be, whether classifiers as unequal as possible to each other would yield better results.

Vorwort

Die Projektarbeit war für uns ein interessanter Ausflug in das Feld des Data-Mining. Wir sind der Überzeugung, dass die Erfahrung im Umgang mit Machine-Learning-Technologien in Zukunft von Nutzen sein wird.

Die Betreuung durch Herr Mark Cieliebak, Herr Oliver Dürr und Herr Fatih Uzdilli war freundschaftlich und förderlich. Wir möchten ihnen dafür danken.

Des Weiteren möchten wir Herr Roger Müller Farguell für die Hilfestellung bei der Strukturierung der Arbeit danken.

Zuletzt gilt unser Dank allen Mitstudenten und Familienmitglieder, die uns geholfen haben diese Arbeit zu verwirklichen.

Lukas Aschwanden und Frederic Saladin

Inhaltsverzeichnis

1 Einleitung....1
1.1 Ausgangslage....1
1.2 Aufgabenstellung....2
1.3 Zielpublikum....2
2 Theoretische Grundlagen....2
2.1 Überblick....2
3 Vorgehen....3
3.1 Versuchsanlage....3
3.2 Herkunft der Daten....3
3.3 Prepräparation der Daten....4
3.4 Hypothesen....5
4 Resultate....6
4.1 Anzahl Klassifikatoren....6
4.2 Qualität der Klassifikatoren....8
5 Diskussion und Ausblick....13
5.1 Weiterführende Ideen....13
6 Verzeichnisse....14
6.1 Literaturverzeichnis....14
6.2 Abbildungsverzeichnis....15
6.3 Glossar....16
7 Anhang....17
7.1 Aufgabenstellung....17
7.2 Datengrundlage....18

1 Einleitung

1.1 Ausgangslage

Es existieren viele verschiedene Algorithmen, im Folgenden als Klassifikatoren bezeichnet, um kurze Texte nach ihrem Sentiment, also ihrer Polarität, zu klassifizieren. Untersuchungen von (Cieliebak et al. 2014) im Bereich der Sentiment-Analyse von Texten haben gezeigt, dass State-of-the-Art-Klassifikatoren im Allgemeinen nicht fähig sind, mehr als 60% der Texte gleich zu klassifizieren, wie es ein Mensch tun würde. Des Weiteren konnten (Cieliebak et al. 2014) beispielhaft zeigen, dass sich die Präzision steigern lässt, wenn ein übergeordneter Meta-Klassifikator die Resultate mehrerer Klassifikatoren auswertet. Außerdem wurde von (Cieliebak et al. 2015) bereits untersucht, welche Kombinationen von Klassifikatoren gute Resultate erzielen, schlüssige Ergebnisse konnten aber noch keine erzielt werden. Somit besteht weiterhin ein Verbesserungspotential. Als Datengrundlage für die Untersuchungen stehen die Sentiment-Vorhersagen von 50 Klassifkatoren zu 1853 Tweets sowie ihre zugehörigen von Menschen bestimmten Sentiments zur Verfügung.

1.2 Aufgabenstellung

Es soll gezeigt werden, dass die Umsetzung eines Meta-Klassifikators mit vielen (50) Klassifikatoren immer sinnvoll ist. Desweiteren sollen Aussagen darüber formuliert werden, wie aus einer grossen Auswahl von Klassifikatoren eine überdurchschnittlich gute Teilmenge gefunden werden kann, um sie mittels eines Meta-Klassifikators zu kombinieren. Zu diesem Zweck sollen Hypothesen entwickelt werden, welche durch Experimente untersucht werden können. Als Grundlage um die benötigten Experimente durchzuführen, muss eine Versuchsanlage implementiert werden. Basierend auf den Resultaten der Untersuchungen sollen allgemeine Empfehlungen zur Auswahl von Klassifikatoren formuliert werden.

1.3 Zielpublikum

Diese Arbeit ist von Interesse, wenn man sich für Meta-Klassifikatoren interessiert. Von konkretem Nutzen sind die vorliegenden Erkenntnisse, bei Data-Mining-Aufgaben, bei denen eine grosse Zahl von Klassifikatoren zur Verfügung stehen.
Alle benötigten Grundkenntnisse werden im folgenden beschrieben verwendete Fachbegriffe werden im Glossar näher beschrieben.

2 Theoretische Grundlagen

Dieses Kapitel stellt das nötige Wissen zum Verständnis der restlichen Arbeit zur Verfügung und setzt sie in einen grösseren Zusammenhang. Es werden grundlegende Informationen und Begriffe des Forschungsbereichs „Data-Mining“ beschrieben.

2.1 Überblick

Wenn es darum geht Datenmengen, die zu gross sind um sie von Hand zu bearbeiten, nach Mustern oder systematischen Zusammenhängen zu durchsuchen, spricht man heute gerne von „Data-Mining“. Die Möglichkeiten dieser jungen IT-Wissenschaft sind ebenso vielfältig, wie die Algorithmen, derer sie sich bedient. Eine dieser Aufgaben ist die Einteilung von Texten nach verschiedenen Kriterien. Einen Computer stellt diese Einteilung vor grosse Schwierigkeiten, da ihm ein wirkliches Verständnis der menschlichen Sprache nicht möglich ist. Hier die verschiedenen Ansätze zu beschreiben, wie versucht wird, dieses Problem zu umgehen, würde zu weit führen. Es gibt vielfältige Klassifikatoren, welche auf verschiedenen Ansätzen beruhen, um es einem Computer zu ermöglichen ohne semantisches Verständnis des Inhalts, eine Einteilung zunehmen zu können. Dazu wird immer eine grosse Menge von Texten mit zugehöriger korrekter Lösung benötigt, mittels dieser versucht der Algorithmus Muster zu finden, um diese dann auf andere Texte anzuwenden, von denen er die Lösung nicht kennt.

"Meta-Learning" ist ein Konzept, das im Data-Mining angewendet wird, um die Vorhersagen mehrerer Klassifikatoren zu kombinieren. Das ist insbesondere dann nützlich, wenn die verschiedenen Klassifikatoren sehr unterschiedlich sind. Eine einfache Variante des Meta-Learning ist „Voting“. Beim Voting zählt der Meta-Klassifikator die Vorhersage der Klassifikatoren und macht die häufigste zu seiner Vorhersage. Daneben gibt es aufwändigere und

leistungsfähigere Meta Learning Methoden, wie Boosting, Bagging oder Random-Forest, welche in der vorliegenden Arbeit verwendet und daher im folgenden auch näher beschrieben wird.

Random-Forest ist ein Algorithmus, der vor allem im Data-Mining eingesetzt wird. Als leichtgewichtiger Meta-Klassifikator ermöglicht er es, die Aussagen mehrerer Klassifikatoren zusammenzufassen, um so eine bessere Vorhersage zu erzielen.

Eine erste Form des Random-Forest wurde von L. Breimann entwickelt, gemeinsam mit A. Cutler entwickelte er auch den Random-Forest, in seiner heute gebräuchlichen weiterentwickelten Form.

Der Ansatz des Random-Forests besteht darin, mithilfe der Aussagen der Klassifikatoren eine große Zahl von Decision-Trees zu bilden, üblicherweise etwa zwischen 200 und 500 Stück. Dabei wird bei jeder einzelnen Frage jedes einzelnen Decision-Trees jeweils zufällig entschieden, welche Aussage berücksichtigt wird. Die Ergebnisse aller gebildeten Decision-Trees werden anschliessend mittels Voting oder Averaging ausgewertet. Um gute Resultate mit dem Random-Forest zu erzielen, ist es wichtig, möglichst unterschiedliche Decision-Trees zu erzeugen (vgl. Livingston 2005, 1-2).

Vorteile des Random-Forests gegenüber anderen Meta-Klassifikatoren sind:

- Gute Qualitätseinschätzung bereits nach dem Trainieren des Random-Forests (OOB-Error).
- Ein Random Forest funktioniert auch mit sehr vielen Features im Verhältnis zur Datenmenge.
- Schlechte Features, die wenig Information tragen, werden automatisch aussortiert.
- Random Forest ist sehr leicht zu implementieren.

3 Vorgehen

3.1 Versuchsanlage

Die Versuchsanlage wurde in Knime, einem graphischen Programmier-Tool, welches sich gut für Data-Mining Aufgaben eignet, implementiert. Die verwendeten Algorithmen stammen aus der Java API „Weka“. Die Resultate von Durchläufen mit dem Meta-Klassifikator werden zunächst als Confusion Matrix abgespeichert. In einem zweiten Schritt werden die gesammelten Ergebnisse dazu benutzt die aufgestellten Hypothesen zu untersuchen.

3.2 Herkunft der Daten

Die verwendeten Daten, stammen aus dem Wettbewerb SemEval-2014. Durchgeführt wurde dieser Wettbewerb von SemEval (Semantic Evaluation), welche regelmäßig Wettbewerbe im Bereich computergestützte Sentimentanalyse von Texten durchführen. SemEval stellte allen Teilnehmern ab dem 15. Dezember 2013 ein Trainingsset, bestehend aus 11'382 Tweets, zur Verfügung. Ab dem 24. März 2014 wurde den Teilnehmern ein Testset, ohne korrekte Sentiments zur Verfügung gestellt. Dieses Testset bestand einerseits aus 3813 Tweets und 2093 Sms, die auch schon im Jahr davor als Testset verwendet wurden, zusätzlich neu hinzugekommenen 1853 Tweets, 86 sarkastische Tweets und 1142 Livejournal-Beiträge. Bis zum 30. März 2014 mussten

die Teilnehmer die Vorhersagen ihres Klassifikators bezüglich des gesamten Testsets einsenden. Als Grundlage für unsere Untersuchungen stehen uns nun von fünfzig Teams, die am SemEval-2014 teilgenommen haben, unter anderem die Vorhersagen ihrer Klassifikatoren zum Testset zur Verfügung. Außerdem sind uns die sogenannten "Goldstandards", also die korrekten Labels des Testsets, bekannt (vgl. Preslav et al. (2013)).

Somit werden für die Untersuchungen folgende Daten verwendet:

Eine einzelne Datei mit dem Goldstandard der Testdaten 2014 vom SemEval. Für jeden Klassifikator der fünfzig Teams jeweils eine Datei, welche eine Beschreibung des Klassifikators und weitere Angaben enthält, eine Datei, welches alle Resultate zu den Testdaten 2014 vom Semeval enthält, sowie eine Datei, welche denScore des Klassifikators enthält, in Form von Präzision, Recall und F1-Score, jeweils aufgeteilt nach der Textquelle: Twitter2013, Twitter2014, Twitter2014Sarcasm, Sms2013, Livejournal2014.

Zur Verifizierung der Versuchsanlage wird auf Daten und Auswertungen der Untersuchungen von (Cieliebak et al. 2014/2015) zurückgegriffen. Diese Daten sind im Kapitel "Tests und Validierung" aufgelistet.

3.3 Prepräparation der Daten

Damit die Wettbewerbsteilnehmer nicht jeweils spezialisierte Klassifikatoren je Textquelle (Twitter, Sms, LiveJournal) verwenden konnten, wurden die Daten aus den verschiedenen Textquellen durchmischt. Das nun zur Verfügung stehende Datenmaterial liegt in Form von unsortierten bereits durch die Wettbewerbsteilnehmer klassifizierten Daten vor. Daher wird zuerst eine Sortierung nach Textquelle vorgenommen.

Partitionierung der Daten in Trainings- und Testset

Die Daten werden partitioniert, um ein Trainings- und ein Testset für den Meta-Klassifikator zu erhalten. Die Partitionierung erfolgt nach dem K-Fold Verfahren:

Es werden 10% der Daten, d.h. ungefähr 180 Tweets, zufällig ausgewählt. Von den verbleibenden Daten werden wiederum 10% zufällig ausgewählt. Dieser Vorgang wird fortgesetzt, bis alle Daten in 10 Partitionen vorliegen. Jede dieser Partitionen wird als Testset benutzt. Die komplementären Daten von jedem Testset, d.h. die anderen 90%, werden als Trainingssets benutzt. Das ergibt 10 Trainings- und Testsets, wobei jedes Set-Paar die gesamten Daten umfasst. Die Absicht hinter diesem Vorgehen ist, für jeden Durchlauf mit dem Meta-Klassifikator jeweils 9 Vergleichs-durchläufe zu haben, um so die statistische Aussagekraft zu verbessern. Die Resultate der einzelnen Durchläufe mit dem Meta-Klassifikator werden, zur weiteren Untersuchung, als Confusion-Matrix abgespeichert.

3.4 Hypothesen

Praktische Limitierungen bezüglich der Suche nach optimalen Teilmengen von Klassifikatoren

Es wäre theoretisch möglich, sämtliche Teilmengen der 50 Klassifikatoren zu bilden und für jede Kombination einen Durchlauf des Meta-Klassifikators mit den Resultaten der ausgewählten Klassifikatoren auszuführen. Allerdings übersteigt die Anzahl an Kombinationen (2^{50}) die zur Verfügung stehenden Ressourcen, Zeit und Rechenleistung, um ein Vielfaches. Deshalb sind Methoden notwendig, die Anzahl zu testender Kombinationen drastisch zu reduzieren. In der vorliegenden Arbeit werden die beiden folgenden Hypothesen näher untersucht.

Quantität der Klassifikatoren

Es soll untersucht werden, welchen Einfluss die Anzahl verschiedener Klassifikatoren auf die Qualität des Random-Forest hat. Voraussichtlich sollten mehr Klassifikatoren zu einer besseren Ergebnis führen, die weitere Verbesserung pro zusätzlichem Klassifikator wird aber wahrscheinlich mit steigender Anzahl kleiner.

Qualität der Klassifikatoren

Es soll untersucht werden, welchen Einfuss die Qualität der einzelnen Klassifikatoren auf die Qualität des Random-Forest hat. Die Performance der einzelnen Klassifikatoren am Semeval 2014 ist bekannt. Es ist anzunehmen, dass Kombinationen von guten Klassifikatoren ein gutes Resultat des Random Forest ergeben werden.

4 Resultate

4.1 Anzahl Klassifikatoren

Um zu untersuchen, wie sich die Anzahl Klassifikatoren auf die Qualität des Random-Forest auswirkt, müssen möglichst viele Testläufe mit unterschiedlicher Anzahl Klassifikatoren durchgeführt werden. Da es nicht möglich ist alle Kombinationen zu testen werden hierzu zunächst zufällige Kombinationen generiert. Für jede mögliche Anzahl Klassifikatoren von 2 bis 48 werden jeweils hundert zufällige voneinander verschiedene Kombinationen generiert. Für die Möglichkeit 49 Klassifikatoren zu kombinieren werden alle 50 Möglichkeiten verwendet und die einzige Möglichkeit alle 50 zu kombinieren wird ebenfalls verwendet.

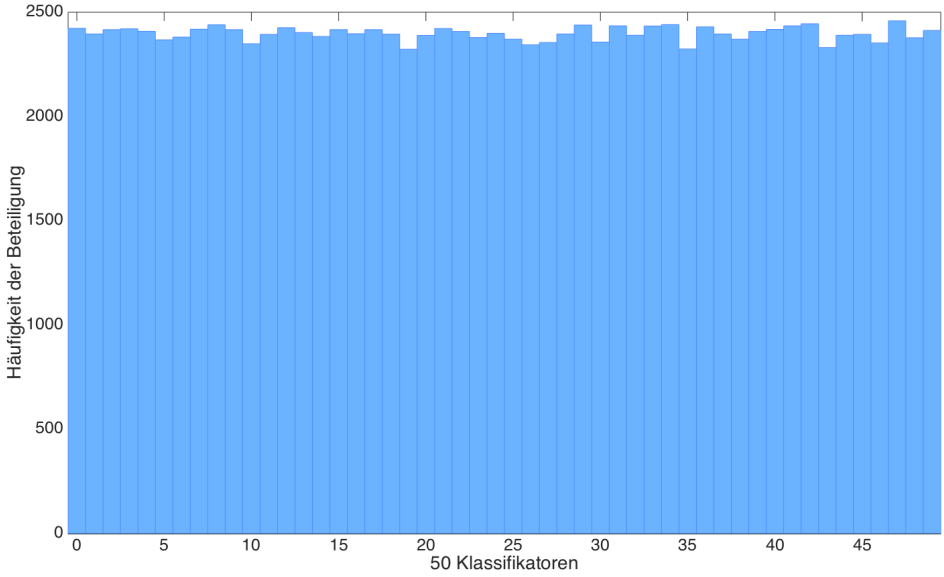

Abbildung 1: Häufigkeit der Beteiligung jedes Klassifikators über alle Resultate des Meta-Klassifikators.

Aus Abbildung 1 ist ersichtlich, dass alle Klassifikatoren insgesamt nahezu gleich oft berücksichtigt wurden. Mittels in Knime erstellten Workflows konnten Testläufe automatisiert durchgeführt werden. In Abbildung 2 sind die Resultate all dieser Testläufe dargestellt.

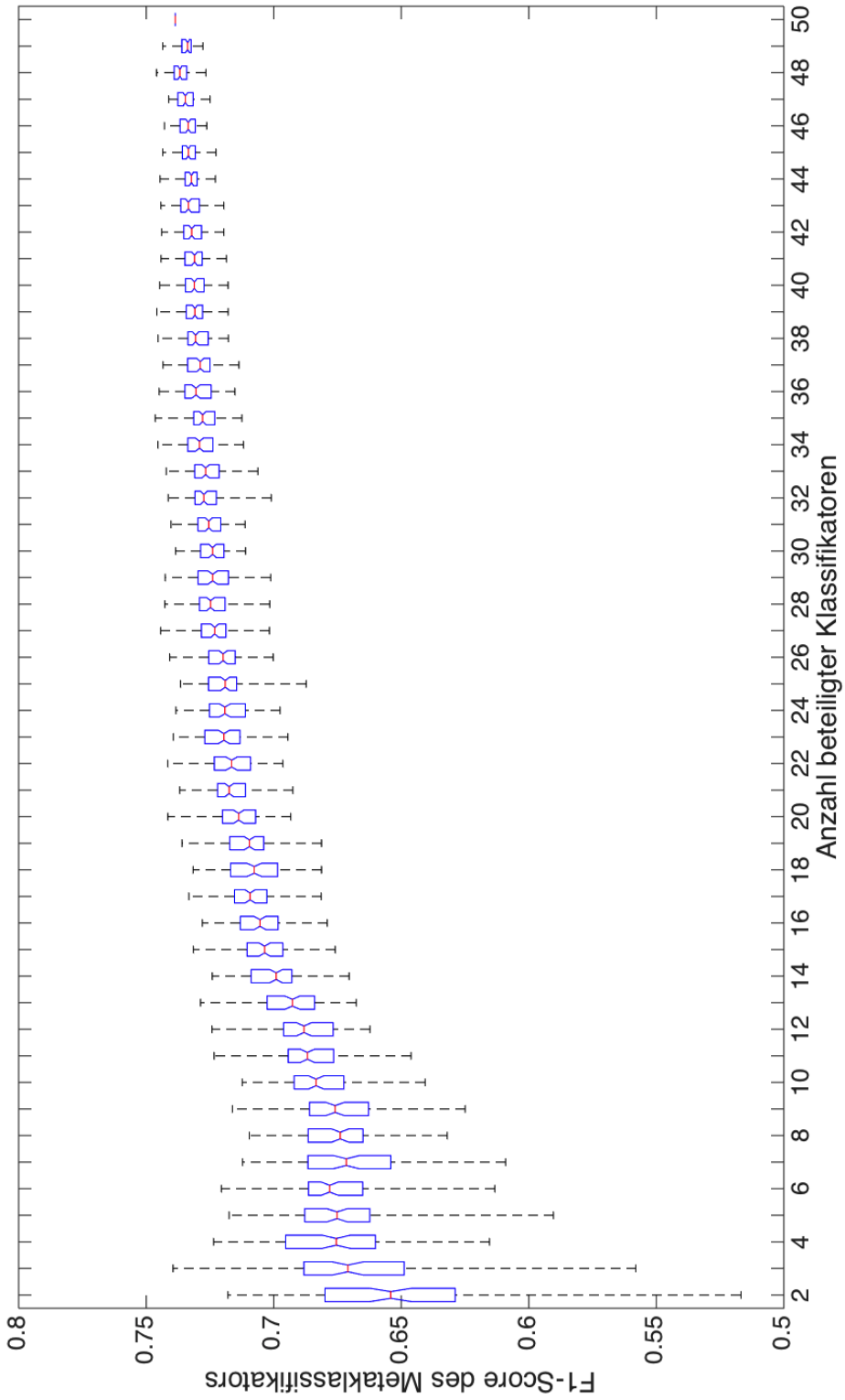

Abbildung 2: F1-Scores aller getesteten Teilmengen von Klassifikatoren aufgeteilt nach der Anzahl beteiligter Klassifikatoren.

Aus Abbildung 2 ist ersichtlich, dass der F1-Score des Meta-Klassifikators mit der Anzahl beteiligter Klassifikatoren ansteigt. Die weitere Verbesserung pro zusätzlichem Klassifikator nimmt mit steigender Anzahl Klassifikatoren ab. Zudem gibt es Ausnahmen, einige Kombinationen von nur drei Klassifikatoren erreichen aussergewöhnlich hohe Präzision während alle fünfzig Möglichkeiten 49 Klassifikatoren zu kombinieren zu ungewöhnlich tiefen Ergebnissen führen.

Fazit

Zusammenfassend lässt sich sagen, dass es sinnvoll ist eine möglichst grosse Anzahl Klassifikatoren zu verwenden, dass jedoch der Zugewinn mit steigender Anzahl bereits verwendeter Klassifikatoren kleiner wird, es muss im Einzelfall entschieden werden wieviel zusätzlicher Aufwand gerechtfertigt werden kann.

4.2 Qualität der Klassifikatoren

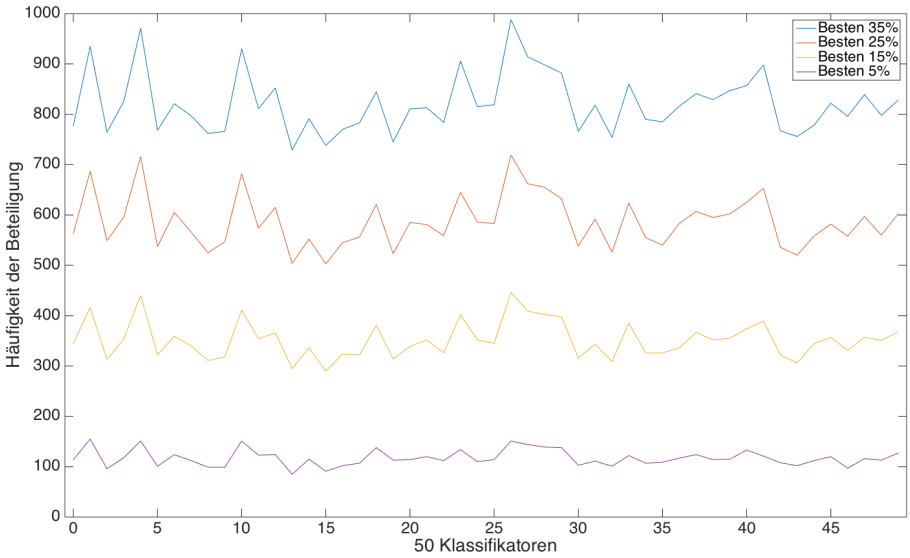

Abbildung 3: Zählungen der Häufigkeiten einzelner Klassifikatoren, mit verschiedenen Prozentsätzen der besten Testläufe pro Anzahl Klassifikatoren.

Um zu verifizieren, ob bei Verwendung von Klassifikatoren mit hohen individuellen F1-Scores jeweils hohe Ergebnisse aus dem Meta-Klassifikator resultieren, wurden die Ergebnisse der Testläufe folgendermassen aufbereitet:

Von den Resultaten von jedem Testlauf mit einer bestimmten Anzahl beteiligter Klassifikatoren wurden verschieden grosse Teilmengen gebildet. Die Teilmengen entsprechen den 5%, 15%, 25% oder 35% der besten Resultaten. Anschliessend wurde anhand dieser Teilmengen grafisch dargestellt, wie häufig die einzelnen Klassifikatoren beteiligt waren. Diese Darstellung ist in Abbildung 3 zu sehen. Auffallend ist, dass bei guten Meta-Klassifikator-Resultaten immer dieselben Klassifikatoren eine höhere Beteiligung haben, dass im Allgemeinen die Beteiligung der Klassifikatoren aber ziemlich gleichmässig ist.

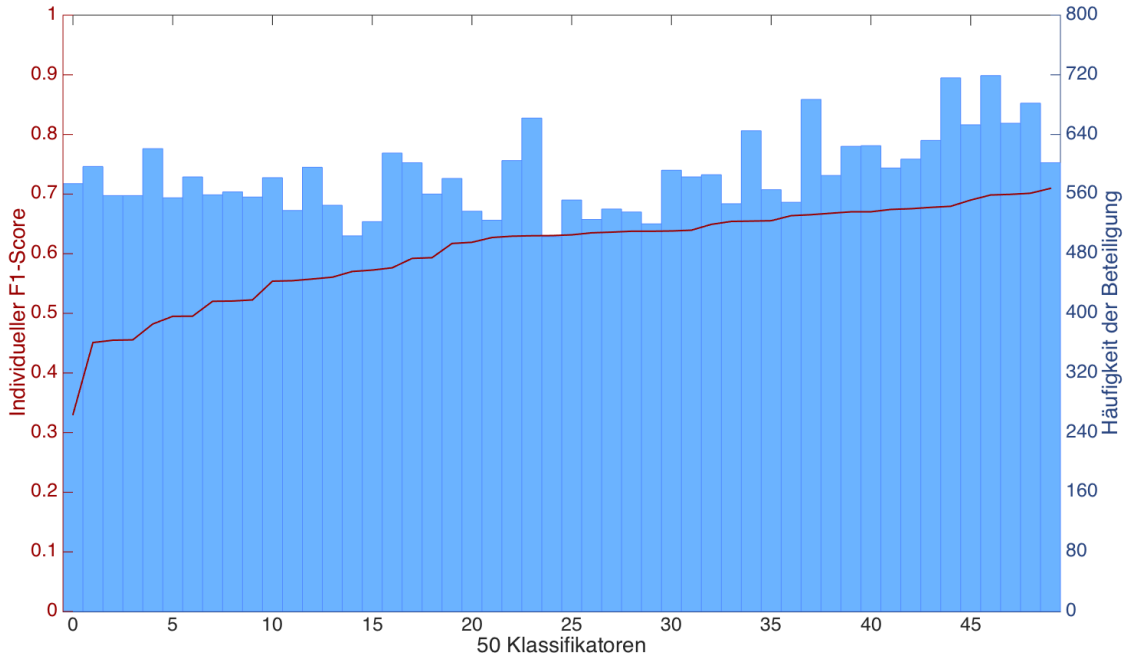

Abbildung 4: Die Häufigkeit jedes einzelnen Klassifikators, in den jeweils besten 25% aller Durchgänge mit bestimmter Anzahl beteiligter Klassifikatoren, im Vergleich zu individuellen F1-Score der Klassifikatoren. Aufsteigend sortiert nach ihrem F1-Score.

Aus Abbildung 4 lässt sich erkennen das nicht jeder Klassifikator mit einem hohen F1-Score häufig an guten Testläufen des Random Forest beteiligt ist. Die 11 Klassifikatoren mit den tiefsten F1-Scores und sogar der Klassifikator mit dem tiefsten F1 score von nur 33.03 sind nicht ungewöhnlich selten beteiligt. Andererseits deutet die höhere Häufigkeit einiger sehr guter Klassifikatoren daraufhin, dass der Random-Forest auf zumindest einige sehr gute Klassifikatoren angewiesen ist um beste Resultate zu erzielen. Dies müsste jedoch noch durch weiterführende Untersuchungen verifiziert werden.

Die Qualität der Ergebnisse eines Random-Forest korreliert entgegen der aufgestellten These nicht mit der Höhe der individuellen F1-Scores der beteiligten Klassifikatoren.

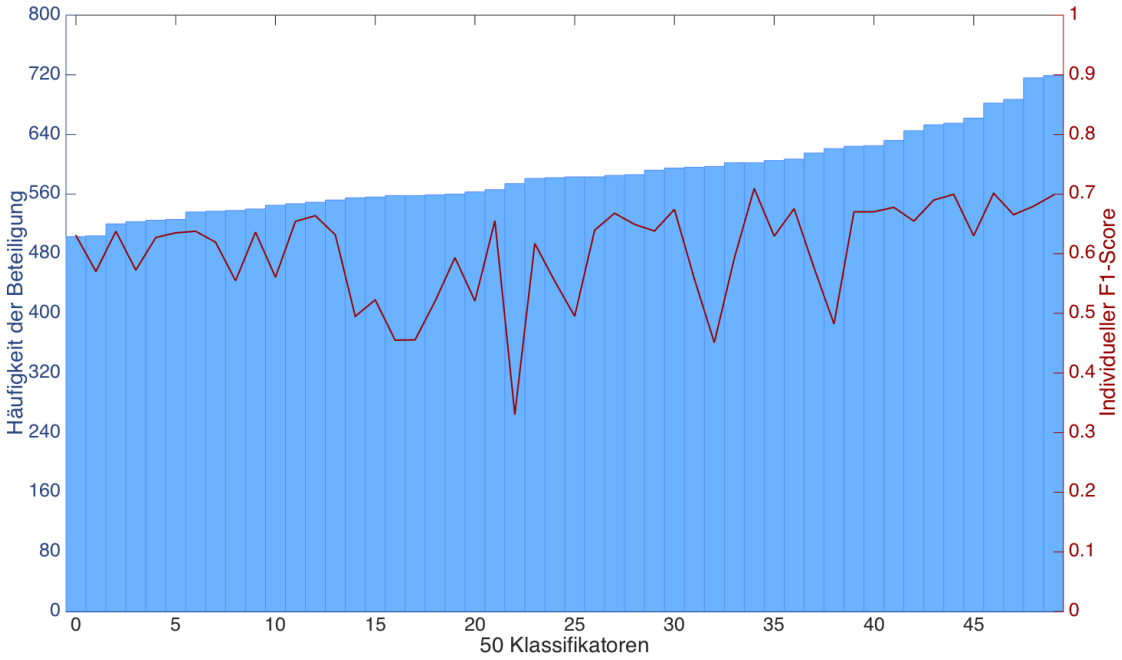

Abbildung 5: Die Häufigkeit jedes einzelnen Klassifikators, in den jeweils besten 25% aller Durchgänge mit bestimmter Anzahl beteiligter Klassifikatoren, im Vergleich zu individuellen F1-Score der Klassifikatoren. Aufsteigend sortiert nach der Häufigkeit der Beteiligung.

Die Unabhängigkeit des F1-Scores von häufiger Vertretung in guten Testläufen sieht man besonders gut in Abbildung 5. Zudem lässt sich hier nochmal sehr schön zeigen, dass die 8 Klassifikatoren mit erhöhter Häufigkeit in den guten Testläufen auch gute F1-Scores haben.

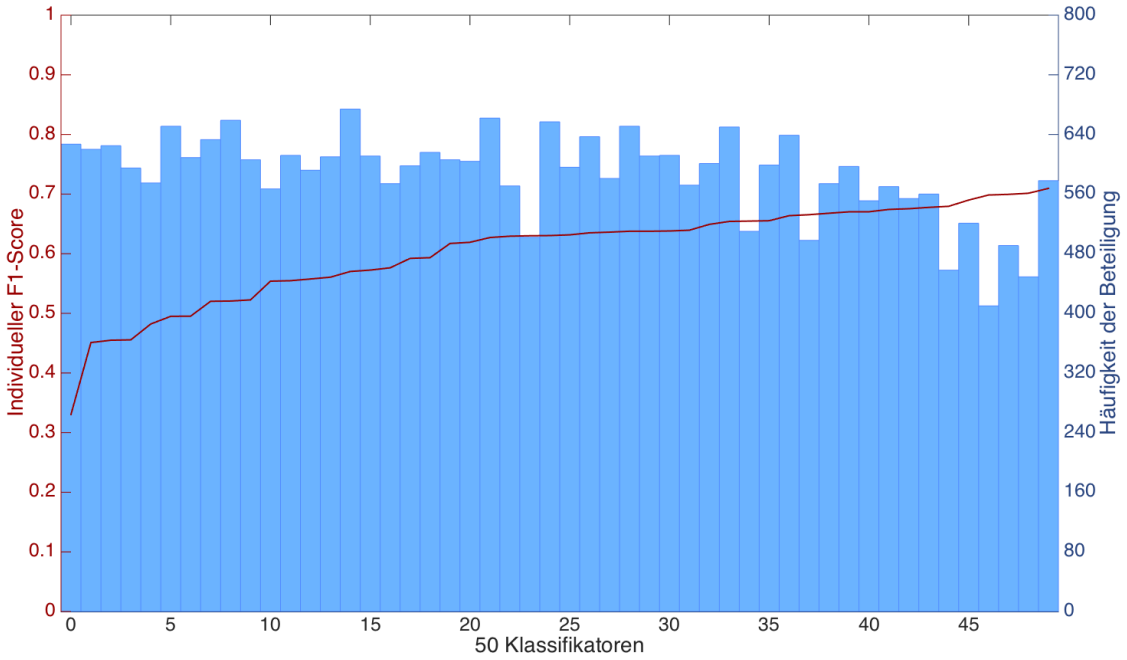

Abbildung 6: Die Häufigkeit jedes einzelnen Klassifikators, in den jeweils schlechtesten 25% aller Durchgänge mit bestimmter Anzahl beteiligter Klassifikatoren, im Vergleich zu individuellen F1-Score der Klassifikatoren. Aufsteigend sortiert nach ihrem F1-Score.

In Abbildung 6 sieht man, wie oft einzelne Klassifikatoren an den schlechtesten Durchgängen des Random-Forest beteiligt waren. Wiederum lässt sich kaum eine Regelmäßigkeit erkennen, ausser einigen Klassifikatoren die auffällig selten beteiligt sind und gleichzeitig tendenziell hohe F1-Scores haben.

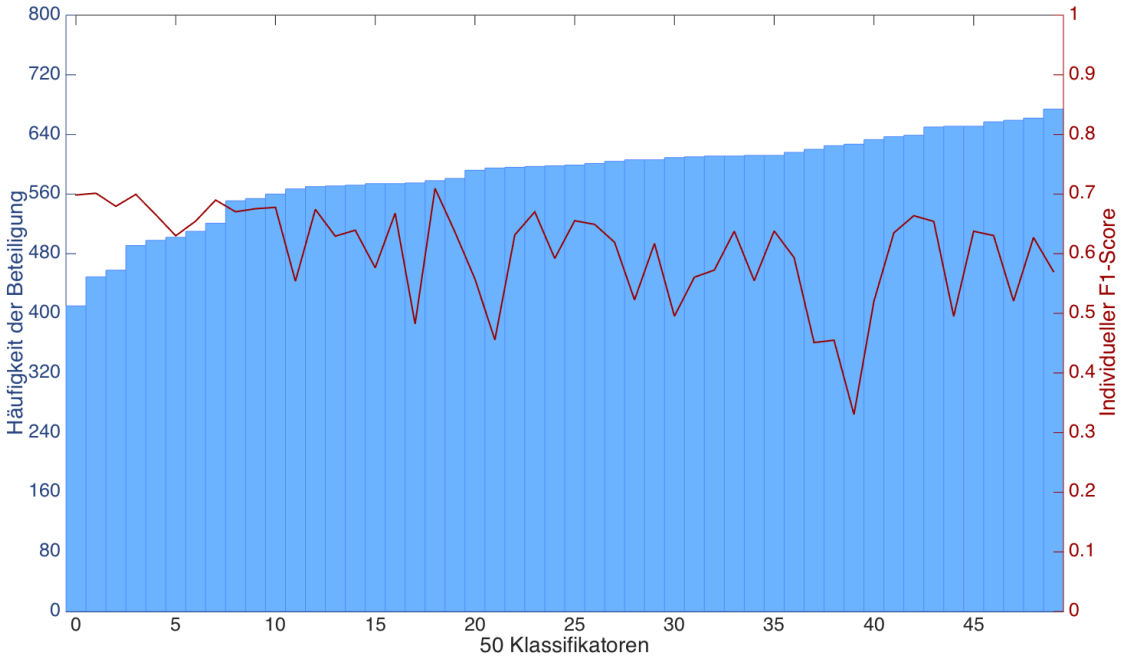

Abbildung 7: Die Häufigkeit jedes einzelnen Klassifikators, in den jeweils schlechtesten 25% aller Durchgänge mit bestimmter Anzahl beteiligter Klassifikatoren, im Vergleich zu individuellen F1-Score der Klassifikatoren. Aufsteigend sortiert nach der Häufigkeit der Beteiligung.

Abbildung 7 verdeutlicht noch einmal die gleichmässige Verteilung der Häufigkeiten, mit Ausnahme einiger Klassifikatoren, die seltener bei den schlechtesten Testläufen vertreten sind und gleichzeitig hohe bis sehr hohe F1-Scores haben.

Fazit

Eine direkte Korrelation zwischen dem F1-Score eines einzelnen Klassifikators und seiner Nützlichkeit für einen Random Forest kann nicht belegt werden. Stattdessen zeigen die erhobenen Daten, dass ein Random-Forest selbst von Klassifikatoren mit sehr tiefem F1-Score profitieren kann. Die Auswahl der Klassifikatoren allein aufgrund ihres F1-Scores ist somit nicht sinnvoll. Insbesondere höhere F1-Scores sind nicht grundsätzlich zu bevorzugen, es sollte jedoch darauf geachtet werden nicht ausschliesslich Klassifikatoren mit tiefem F1-Score zu benutzen.

5 Diskussion und Ausblick

5.1 Weiterführende Ideen

Die durchgeführten Untersuchungen könnten weitergeführt und vertieft werden, um das Auswahlverfahren von Klassifikatoren, welche man mittels eines Meta-Klassifikators kombinieren sollte, weiter zu verbessern oder den Einsatzbereich zu erweitern.

Kombinationen von Klassifikatoren aufgrund der Ähnlichkeit der Aussagen der einzelnen Klassifikatoren vergleichen

Anstatt zufällige Kombinationen von Tools zu verwenden, könnte man die Klassifikatoren zunächst aufgrund ihrer Ähnlichkeit in Cluster aufteilen. Dadurch könnte man Vergleiche ziehen zwischen Versuchen mit Klassifikatoren aus nur einem und aus vielen verschiedenen Clustern. Clustering wird häufig im Datamining verwendet, daher stellen die verwendeten Tools, wie Weka, entsprechende Algorithmen zur Verfügung. Klassifikatoren aus verschiedenen Clustern würden dabei voraussichtlich bessere Ergebnisse erzielen.

Meta-Klassifikator auswechseln

Der Random-Forest ist ein leicht zu implementierender Meta-Klassifikator, der gerne im Datamining eingesetzt wird. Ein aufwändigerer Meta-Klassifikator könnte bessere Ergebnisse erzielen. Aufgrund eines neuen Meta-Klassifikators könnten neue Fragestellungen entstehen.

Vom Random-Forest anzeigen lassen welche Tools wenig Information beitragen

Für einen Klassifikator, somit auch für einen Meta-Klassifikator ist die Menge an Informationen, die die übergebenen Features tragen, von großer Bedeutung. Wenn man Klassifikatoren, die wenig Informationen beitragen, weg lässt, würde man eine bessere Performance des Random-Forest erwarten. Die Implementierung des Random-Forest in Weka bietet die Möglichkeit, zwei verschiedene Ausgaben zu erzeugen, welche beide Aufschluss darüber geben können, welche Features viel Information beitragen.

Random-Forest Einstellungen optimieren

Beim Random-Forest lässt sich die Anzahl der Bäume, die er erzeugen soll, manuell einstellen. L. Breimann, der Erfinder des Algorithmus empfiehlt etwa 200 bis 500. Innerhalb dieses empfohlenen Bereichs ließe sich eine Einstellung finden, mit welcher der Random-Forest möglichst gute Ergebnisse erzielt. Hierzu müssten jeweils identische Durchläufe durchgeführt werden, bei denen nur die Anzahl der Bäume variiert. Wir gehen davon aus das die Anzahl der Bäume einen vergleichsweise geringen Einfluss auf die Performance des Random-Forest hat. Um eine zuverlässige Aussage darüber zu machen, müsste man sehr viele Durchläufe durchführen können.

6 Verzeichnisse

6.1 Literaturverzeichnis

Cieliebak, Mark et al. (2014): Meta-Classifiers Easily Improve Commercial Sentiment Detection Tools.

In: *Proceedings of the 9th edition of the Language Resources and Evaluation Conference* (LREC) (Reykjavik, Iceland) 26-31.05.2014 , 3100-3104

Cieliebak, Marc et al. (2015): *JOINT_FORCES: Unite Competing Sentiment Classifiers with Random Forest*

URL: http://www.aclweb.org/anthology/S14-2062

[Stand: 29.08.2014]

Livingston, Frederick (2005): *Implementation of Breiman's Random Forest Machine Learning Algorithm*

URL: http://www4.ncsu.edu/~fjliving/docs/JournalPaper_Livingston.pdf

[Stand: 15.10.2014]

Preslav, Nakov et al. (2013): SemEval-2013 Task 2: Sentiment Analysis in Twitter.

In: *Proceedings of the International Workshop on Semantic Evaluation* (SemEval- 2013)(USA, Georgia), 14-15.06.2013, 312-320

Silipo, Rosaria et al. (2012): *Creating Usable Customer Intelligence from Social Media Data: Clustering the Social Community*

URL: https://www.knime.org/files/fillpdf/knime_social_media_data_clustering_whitepaper.pdf

[Stand: 22.10.2014]

Silipo, Rosaria et al. (2012): *Creating Usable Customer Intelligence from Social Media Data: Network Analytics meets Text Mining*

URL: https://www.knime.org/files/knime_social_media_white_paper.pdf

[Stand: 22.10.2014]

Xia, Rui et al. (15.3.2011): Ensemble of feature sets and classification algorithms for sentiment classification

In: *Information Sciences*, Bn. 181, Issue 6, 2009, S. 1138-1152

6.2 Abbildungsverzeichnis

Abbildung 1: Häufigkeit der Beteiligung jedes Klassifikators über alle Resultate des Meta-Klassifikators....6

Abbildung 2: F1-Scores aller getesteten Teilmengen von Klassifikatoren aufgeteilt nach der Anzahl beteiligter Klassifikatoren....7

Abbildung 3: Zählungen der Häufigkeiten einzelner Klassifikatoren, mit verschiedenen Prozentsätzen der besten Testläufe pro Anzahl Klassifikatoren....8

Abbildung 4: Die Häufigkeit jedes einzelnen Klassifikators, in den jeweils besten 25% aller Durchgänge mit bestimmter Anzahl beteiligter Klassifikatoren, im Vergleich zu individuellen F1-Score der Klassifikatoren. Aufsteigend sortiert nach ihrem F1-Score....9

Abbildung 5: Die Häufigkeit jedes einzelnen Klassifikators, in den jeweils besten 25% aller Durchgänge mit bestimmter Anzahl beteiligter Klassifikatoren, im Vergleich zu individuellen F1-Score der Klassifikatoren. Aufsteigend sortiert nach der Häufigkeit der Beteiligung....10

Abbildung 6: Die Häufigkeit jedes einzelnen Klassifikators, in den jeweils schlechtesten 25% aller Durchgänge mit bestimmter Anzahl beteiligter Klassifikatoren, im Vergleich zu individuellen F1-Score der Klassifikatoren. Aufsteigend sortiert nach ihrem F1-Score....11

Abbildung 7: Die Häufigkeit jedes einzelnen Klassifikators, in den jeweils schlechtesten 25% aller Durchgänge mit bestimmter Anzahl beteiligter Klassifikatoren, im Vergleich zu individuellen F1-Score der Klassifikatoren. Aufsteigend sortiert nach der Häufigkeit der Beteiligung....12

6.3 Glossar

Confusion Matrix

Eine "Confusion Matrix" ist eine gebräuchliche Darstellung für die Qualität einer Klassifizierung. Es ist eine Tabelle, in der die Spalten die vorhergesagte Klasse angeben und die Kolonnen die tatsächliche Klasse.

Decision Tree

Eine häufig verwendete spezielle Art des Klassifikators ist der „Decision Tree“. Ein Decision Tree, also ein Entscheidungsbaum, ist eine Anordnung von hierarchisch aufeinanderfolgenden Entscheidungen. Abhängig von der Antwort auf die erste Frage folgt eine neue Frage, von deren Antwort wieder die nächste Frage abhängt, falls es aufgrund einer Antwort keine weitere Frage gibt, ist eine der möglichen Klassierungen erreicht.

Feature

Als Feature bezeichnet man im Data-Mining die Merkmale von Daten, aufgrund derer Klassifikatoren die Daten einzuteilen versuchen. Für einen Meta-Klassifikator bedeutet dies, dass die Aussagen der einzelnen Klassifikatoren als Features betrachtet werden.

Goldstandard

Als "Goldstandard" bezeichnet SemEval die Angaben, die von Menschen festgelegt werden, ob das Sentiment eines Texts positiv, neutral oder negativ ist. Mit diesen Goldstandards werden die Aussagen von Klassifikatoren verglichen, um zu entscheiden, ob sie das Sentiment korrekt klassiert haben.

K-Fold

"K-Fold" ist ein Testverfahren der Statistik, um die Präzision eines Models festzustellen. Dabei wird die Datenmenge in möglichst gleich große k Teilmengen aufgeteilt wird. Danach werden k Programmdurchläufe gemacht wobei jeweils eine andere der k Teilmengen als Testmenge und jeweils alle anderen Teilmengen als Trainingsmenge verwendet werden. Mit der durchschnittlichen Fehlerquote aller Testdurchläufe lässt sich so eine realistische Abschätzung der Fehlerquote des Models erreichen.

Klassifikator

Als "Klassifikator" wird ein Programm bezeichnet, dass versucht eine Menge von Dingen aufgrund ihrer Merkmale in verschiedene "Schubladen" einzuteilen.

In der vorliegenden Arbeit geht es beispielsweise darum, kurze Texte aufgrund ihres Inhalts in die drei möglichen "Schubladen" positives Sentiment, neutrales Sentiment oder negatives Sentiment einzuteilen.

Im Gegensatz hierzu gibt es auch Klassifikatoren, bei denen die Schubladen erst im Verlauf des Programms durch das Programm selber, aufgrund von Häufungen von Merkmalen, festgelegt werden.

Meta-Klassifikator

Ein "Meta-Klassifikator" kann verwendet werden, um die Einteilung mehrerer anderer Klassifikatoren zu kombinieren und basierend darauf eine neue Einteilung zu machen. Dadurch kann häufig eine korrektere Einteilung erreicht werden.

Out-of-Bag-Error

Als Out-of-Bag-Error wird sowohl das Verfahren als auch das Resultat genannt, mit welchem ein Random-Forest bereits während seiner Erzeugung eine Abschätzung seiner Qualität machen kann. Hierzu werden bei jedem Baum jeweils die Daten, welche bei seinem Training weggelassen wurden, als Testset verwendet. Das durchschnittliche Resultat dieser Testläufe ist der Out-of-Bag-Error der in der Regel eine sehr gute Abschätzung der Qualität des Random-Forest abgibt.

7 Anhang

7.1 Aufgabenstellung

Kurzbeschreibung

In dieser Projektarbeit entwickeln und evaluieren Sie einen Meta-Klassifikator um aus vielen guten und mittelprächtigen Sentiment-Analyse-Klassifikatoren einen besseren zu konstruieren. Dabei werden Sie Methoden aus dem Data-Mining und Machine-Learning einsetzen.

Hintergrund

Mit Sentiment Analyse lassen sich Texte automatisch in positiv, negativ und neutral klassifizieren. Die Anwendungsfelder sind vielfältig, gerade im Bereich Social Media: Vorhersage von Börsenkursen, Wahlprognosen, Auswertung von Marketing-Kampagnen, Verbesserung des Kundendienstes etc. Über 7000 wissenschaftliche Artikel sind in den letzten Jahren dazu geschrieben und Hunderte von Startups gegründet worden. Allerdings lässt die Erkennungsrate der Verfahren immer noch zu wünschen übrig. Sie liegt zurzeit bei ca. 60% - das heisst 4 von 10 Dokumenten werden falsch klassifiziert.

Wir haben in einem aktuellen Forschungsprojekt gezeigt [1], dass man die Erkennungsrate erhöhen kann, indem man einen Meta-Klassifikator verwendet. Bisher standen uns dafür allerdings nur einzelne Klassifikatoren zur Verfügung. Seit kurzen verfügen wir über einen Datensatz mit den Ergebnissen von über 50 Klassifikatoren. Dieser Datensatz eröffnet eine Vielfalt von neuen Analysemöglichkeiten, die wir in dieser PA ausloten möchten.

Aufgabe

Ihre Aufgabe in dieser PA ist es, zunächst die Methoden aus [1] auf den neuen Datensatz anzuwenden und auszuwerten. Anschliessend entwickeln Sie auf Basis der Ergebnisse eigene Vorschläge für neue Analysen, Methoden und Auswertungen, und führen diese durch.

Wir sind überzeugt, dass man am Ende der PA die Ergebnisse in einem wisschenschaftlichen Paper publizieren kann.

[1] Cieliebak, Mark; Dürr, Oliver; Uzdilli, Fatih (2014). Meta-Classifiers Easily Improve Commercial Sentiment Detection Tools. Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC 2014)

7.2 Datengrundlage

SemEval veranstaltete 2014 den "SemEval 2014"-Wettbewerb. Die folgende Auflistung zeigt welche Datensätze es vor und während des Wettbewerbs gab und wozu sie jeweils verwendet wurden.

Trainings-Datensätze

Diese Datensätze waren dazu gedacht, einen Machine-Learning-Algorithmus zu trainieren.

- Datensatz: `train-2014` beinhaltet `Tweet-Nachrichten` und `wahre Sentiments`.
- Datensatz: `dev-2014` beinhaltet `Tweet-Nachrichten` und `wahre Sentiments`.

Tuning-Datensätze

Diese Datensätze durften benutzt werden, um bessere Einstellungen für die Parameter der Algorithmen zu finden. Nachdem der Algorithmus trainiert wurde, lässt man den Algorithmus Voraussagen für diese Datensätze machen. Danach ändert man Parameter und lässt den Algorithmus erneut Voraussagen machen. Diesen Vorgang wiederholt man beliebig viele Male und versucht so Parameter zu finden, die die Präzision der Voraussagen erhöht.

Dieser Vorgang ist fast derselbe wie der Vorgang des Trainierens. Je nachdem, wie der Algorithmus aufgebaut ist, kann es sein, dass keine Unterscheidung mehr zwischen Training und Tuning gemacht werden kann.

- Datensatz: `test-tweets-2013`beinhaltet `Tweet-Nachrichten` und `wahre Sentiments`. Dieser Datensatz entspricht dem Tweet-Ausschnitt des Wettbewerb-Datensatzes aus dem vorangegangenen Jahr 2013.
- Datensatz: `test-sms-2013` beinhaltet `SMS-Nachrichten` und `wahre Sentiments`. Dieser Datensatz entspricht dem SMS-Ausschnitt des Wettbewerb-Datensatzes aus dem vorangegangenen Jahr 2013.

Wettbewerb-Datensatz

Für den Wettbewerb gab es einen einzigen Datensatz mit vielen Nachrichten. Der Wettbewerb bestand darin, möglichst präzise Sentiment-Voraussagen mittels der trainierten und getunten Algorithmen zu machen. Datensatz: `competition` beinhaltet `Nachrichten`. Tatsächlich stammten die Nachrichten im `competition`-Datensatz aus verschiedenen Quellen/Kategorien, allerdings kannten die Wettbewerbsteilnehmer diese Unterscheidung nicht, bzw. für sie schienen alle Nachrichten aus einer einzigen unbekannten Quelle/Kategorie zu stammen. Diese während des Wettbewerbs unbekannten Quellen/Kategorien sind hier aufgelistet:

- `test-tweets-2014` beinhaltet bis dahin unbekannte `Tweet-Nachrichten`.
- `test-tweets-sarcasm-2014` beinhaltet bis dahin unbekannte `Tweet-Nachrichten`, deren Inhalt sarkastisch ist.
- `test-livejournal-2014` beinhaltet bis dahin unbekannte `Live-Journal-Nachrichten`. Zu beachten: Für diese Quelle/Kategorie von Nachrichten gab es kein Äquivalent unter den Trainings-Datensätzen.
- `test-tweets-2013` ist derselbe Datensatz wie `test-tweets-2013` bei den Tuning-Datensätzen. Er wurde hinzugefügt, um Teilnehmer zu ermitteln, die illegal auf diesen Datensätzen trainiert haben.
- `test-sms-2013` ist derselbe Datensatz wie `test-sms-2013` bei den Tuning-Datensätzen. Er wurde hinzugefügt, um Teilnehmer zu ermitteln, die illegal auf diesen Datensätzen trainiert haben.