Master-Thesis

Thunderbird-Plugin zur Erkennung von anhangverdächtigen E-Mails

TECHNISCHE UNIVERSITÄT DARMSTADT

Marco Ghiglieri

Fachgebiet Knowledge Engineering
Prof. Dr. Johannes Fürnkranz

29. Januar 2009

Ehrenwörtliche Erklärung

Hiermit versichere ich, die vorliegende Masterarbeit ohne Hilfe Dritter und nur mit den angegebenen Quellen und Hilfsmitteln angefertigt zu haben. Alle Stellen, die aus den Quellen entnommen wurden, sind als solche kenntlich gemacht worden. Diese Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Darmstadt, den 29. Januar 2009

Danksagung

Ich bedanke mich herzlich bei Professor Johannes Fürnkranz, der mir diese interessante Arbeit erst ermöglicht hat, und sie bestens betreute.
Meinen Eltern, meiner Schwester und meinen Freunden ist an dieser Stelle mein tiefster Dank ausgedrückt. Ohne eure Unterstützung in allen Lebenslagen wäre es gar nicht zu diesem Studium und damit dieser Arbeit gekommen.
Ganz besonderer Dank gilt meiner Freundin Jane Elsemüller, die mir in allen Angelegenheiten beistand und viele Tage damit verbrachte, Fehler in der Arbeit zu korrigieren. Auch Tino Flächsenhaar möchte ich für hilfreiche Ratschläge und das Korrekturlesen danken. Abschließend danke ich Stephan Adam, der seine E-Mails mehrere Male einer Analyse unterzog und seinen Rechner teilweise eine Nacht laufen lassen musste.

Ich wünsche allen Nutzern des AttachmentCheckers viel Erfolg und weniger vergessene Attachments!

Inhaltsverzeichnis

Kapitel 1

Einleitung

Jeder, der viele E-Mails schreibt, versendet und empfängt, hat wahrscheinlich auch schon einmal eine E-Mail versendet oder erhalten, die einen Anhang — im folgenden auch Attachment genannt — haben sollte, aber keinen hatte. Vergessene Attachments können sehr unangenehm sein, zum Beispiel, wenn es sich um E-Mails an den Vorgesetzten, E-Mails im Rahmen von Bewerbungen oder um E-Mails an Freunde handelt.
Rechnen wir diesen Sachverhalt einmal hoch. Es werden weltweit ungefähr 210 Milliarden E-Mails pro Tag verschickt. Davon sind etwa zwei Drittel Spam, die wir abziehen können. Bleiben noch ungefähr 70 Milliarden E-Mails pro Tag mit „sinnvollen“ Daten übrig.[Tsc08]
Nehmen wir nun an, dass zehn Prozent aller verschickten E-Mails eines E–Mail–Nutzers Attachments enthalten, so werden also täglich sieben Milliarden E-Mails mit Attachments versandt. Gehen wir außerdem davon aus, dass ein E–Mail–Nutzer in einem Prozent der Fälle, in denen er ein Attachment schicken möchte, dieses vergisst, bedeutet dies, dass täglich 70 Millionen E-Mails verschickt werden, bei denen das Attachment fehlt!
Jetzt gibt es mehrere Möglichkeiten, welche Reaktionen dies beim Sender oder Empfänger auslösen könnte. Bemerkt der Sender direkt, dass er an der E-Mail ein Attachment vergessen hat, verschickt er gleich im Anschluss eine weitere E-Mail mit dem fehlenden Attachment. Es ist nichts weiter passiert, außer dass der Empfänger zwei E-Mails erhält. Merkt der Sender den fehlenden Anhang nicht frühzeitig, müsste der Empfänger das Attachment anfordern, falls er weiß, dass ein Attachment fehlt. Der schlechteste Fall wäre, dass beide Parteien nichts von dem Mißgeschick bemerken und die Daten nie übermittelt werden. Dies kann für den Sender, je nach Verwendungszweck der E-Mail, eine sehr peinliche Situation hervorrufen.
Mit dem AttachmentChecker, so heißt das Plugin dieser Arbeit, soll das Problem möglichst minimiert werden.

1.1 Problemstellung

Die Problemstellung dieser Arbeit besteht darin, ein Mozilla Thunderbird-Plugin zu entwickeln, dass durch geeignete Verfahren des maschinellen Lernens E-Mails klassifiziert. Das Klassifizierungsproblem muss die Frage, ob die E-Mail vermutlich ein Attachment haben sollte, lösen.
Sollte eine E-Mail als E-Mail mit Attachment klassifiziert und kein Attachment angehängt worden sein, soll das Plugin den Verfasser der E-Mail auf das wahrscheinlich vergessene Attachment hinweisen. Die Frage soll mittels einem Popup von Thunderbird gestellt werden, welches erscheint, wenn der Verfasser die E-Mail endgültig verschicken möchte, also beim Klick auf „Senden“.
Der Optimalfall wäre ein Plugin, dass alle vergessenen Attachments findet und wenige E-Mails falsch klassifiziert.

1.2 Unterschiede Spam- und Attachment-Erkennung

Erfahrungen des Alltags zeigen, dass die Masse der Spam-E-Mails in den letzten Jahren bedrohlich gewachsen ist. So mussten Verfahren entwickelt werden, die eine Klassifizierung der ungewünschten E-Mails ermöglichten. Anfangs versuchte man, mit einfachen Wortlisten das Spam-Problem einzudämmen. In solchen Wortlisten sind Worte enthalten, die häufig in Spam-E-Mails vorkommen. Mit diesen Listen kann erkannt werden, ob es sich um Spam handelt.
Das Wortlisten-Verfahren ist den Spam-Versendern allerdings bestens bekannt und so werden Spam-E-Mails dahingehend modifiziert, dass eine Klassifizierung mit Hilfe einfacher Wortlisten nahezu unmöglich ist. Die Reaktion auf diese Anpassung war eine Hinzuziehung weiterer Erkennungsmerkmale, wie die Großschreibung der meisten Worte, die Sprache und die Serverprüfung. Tatsächlich erkennt man dennoch nur knapp 70 % der Spam-E-Mails. Nachvollziehbar sinkt die korrekte Erkennung ständig, da die Spam-E-Mails sich an gewöhnliche E-Mails anpassen. Dennoch sind einige solcher Filter noch immer in veralteten Softwarepaketen im Einsatz. Das Schlimmste an diesem Ansatz ist aber, dass es sehr viele Fehlklassifikationen gibt, was dazu führt, dass die Spam-E-Mails und die normalen E-Mails nochmals vom E-Mail-Nutzer durchgeschaut werden müssen. Dies bringt natürlich keinen Vorteil im Gegensatz zur manuellen Filterung.
Der Einsatz von statistischen Verfahren, wie Naive Bayes oder Paul Grahams Algorithmus, bringt einen merklichen Fortschritt in der Spambekämpfung. Die Spamfilter sind mittlerweile so gut, dass sie bei einer Erkennungsrate von über 99 % mit weit weniger als einem Prozent Fehlklassifizierungen liegen.[Gra03]
Der derzeitige Trend zeigt eine immer schwierigere Spam-Klassifizierung, da sich Spam-E-Mails immer weniger von normalen E-Mails unterscheiden. Ein „Wettrüsten“ zwischen Spam-Versendern und Erkennungsfiltern ist zu beobachten.
Die hier beschriebenen Verfahren laufen zum größten Teil im E-Mail-Programm des

Nutzers ab. Darüber hinaus, werden globale serverseitige Filter eingesetzt. In diesen globalen Filtern, werden Merkmale abgefangen, die nicht benutzerspezifisch sind, damit bekommt der Benutzer die vermutlichen Spam-E-Mails nicht mehr auf den eigenen Rechner. Unter die nicht benutzerspezifischen Merkmale fallen vor allem Sperrlisten, die komplette IP-Adressen oder sogar der IP-Bereich aus dem der Server stammt, der die E-Mail zugestellt hat. Über globale Filter versucht man, Server von Anfang an auszuschließen, weil über diese nur Spam versandt wird. Dies geschieht im Regelfall mit den IP-Adressen der Versender, die derzeit verstärkt überprüft werden. Ist ein Server einmal auf einer Sperrliste kann er von dieser nicht ohne Weiteres wieder gelöscht werden, was zur Folge hat, dass von diesem Server keine E-Mails mehr versendet werden können.
Zusammenfassend kann gesagt werden, dass Spam-E-Mails derzeit kein allzu großes Problem darstellen. Lediglich eine Fehlklassifizierung kann unangenehm für den Nutzer sein: Eine E-Mail, die kein Spam ist, aber im Spam-Ordner landet, wird womöglich niemals vom Benutzer gesehen. Eine Spam-E-Mail, die hingegen im Posteingang landet, ist weniger schlimm.
Beim Attachment-Erkennungsproblem kann man ähnliche Filter für die Erkennung einsetzen, da es sich auch hier um ein Zweiklassenproblem handelt. Die beiden Klassen sind hier „Attachment“, in der E-Mails mit Attachment und „Noattachment“, in der E-Mails ohne Attachment eingeordnet werden. Beim Spam Problem handelt es sich hingegen um „Spam“ und „Ham“, wobei „Ham“ die Klasse der erwünschten E-Mails darstellt.
Die einfachste Methode ist, eine Wortliste durch den Benutzer anzulegen, die häufige Worte, die in Attachment-E-Mails vorkommen auflistet. Ist ein Wort in der Liste, ist die E-Mail anhangverdächtig. Derzeit gibt es einige Programme, die dieses Verfahren einsetzen:

- Attachment Reminder für Thunderbird [PK08]
- Check and Send für Thunderbird [H.O08]
- Gmail Attachment Reminder [gma07]

Wie auch bei der Spam-Erkennung gibt es hier einige Probleme, die adressiert werden müssen. Diese Verfahren erkennen nur eine Teilmenge der E-Mails mit Attachment. Wählt ein Benutzer die falschen Worte für die Wortliste, funktioniert der Filter nicht.
Ein Unterschied zum Spam-Problem und damit ein deutlicher Vorteil bei der Attachment-Erkennung ist, dass es sich bei den Attachment-E-Mails um benutzerspezifische E-Mails handelt. Die E-Mails ändern also ihre charakteristischen Merkmale nicht, da beide E-Mail-Klassen vom Verfasser kommen und damit eine dynamische Anpassung der Wortliste nicht notwendig ist.
Um die manuelle Pflege der Wortliste zu unterbinden, benötigt man einen „Lerner“, wie den Naive Bayes oder Paul Graham Algorithmus. Ein Nachteil ist, dass eine Erkennung anhand von anderen Merkmalen, wie Server oder Sprache, nicht möglich ist. Attachment-E-Mails wurden gewöhnlich noch nicht versandt und besitzen somit auch

noch keine Serverinformationen. Die Sprache soll gelernt werden und kann somit nicht zur Unterscheidung dienen.
Beim Attachment-Problem handelt es sich dennoch um ein schwierigeres Zweiklassenproblem, da die E-Mails sich nur gering unterscheiden. Aber im Gegensatz zur Spam-Erkennung ist die Fehlklassifizierung einer E-Mail weniger kritisch als bei der Einordnung in „Ham“ und „Spam“, weil hier kein Schaden angerichtet wird. Natürlich ist es nutzerabhängig, welcher Fall E-Mail-Attachmment nicht erkannt oder E-Mail ohne Attachment gemeldet als schlimmer einzustufen ist.

1.3 Aufbau der Arbeit

Die Arbeit teilt sich in einen praktischen und schriftlichen Teil. Der schriftliche Teil basiert auf dem AttachmentChecker in der Version 0.36, der auf meiner Webseite und der beiliegenden CD dieser Arbeit enthalten ist.[Ghi]
Der schriftliche Teil gliedert sich inhaltlich in sechs Kapitel. Das erste Kapitel stellt eine kurze Einführung in das Thema dar. Unter anderem gibt es eine Gegenüberstellung des bekannten Spam-Problems und der Attachment-Erkennung. Das zweite Kapitel beschäftigt sich mit den theoretischen Hintergründen dieser Arbeit. Beginnend mit einer Erklärung, was maschinelles Lernen ist, gefolgt von der Definition des Attachment-Problems bis zu den konkreten Algorithmen ist in diesem Kapitel alles enthalten. Kapitel 3 stellt die praktische Seite der Arbeit da und zeigt, wie das Plugin umgesetzt wurde. Besonderheiten, die nicht — wie in Kapitel 2 beschrieben — umgesetzt sind, werden dort näher erläutert. Im vierten Kapitel wird dargestellt, ob die Algorithmen aus Kapitel 2 auch funktionieren. Dort werden einige Testergebnisse näher erläutert. Das fünfte Kapitel stellt das Fazit und weitere Möglichkeiten, diese Arbeit zu verbessern vor. Das sechste Kapitel beinhaltet den Anhang, der alle Grafiken, alle Testergebnisse, Literaturangaben und das Abbildungsverzeichnis enthält.
Optisch wird unterschieden zwischen Algorithmen, Beispielen und Abbildungen. Diese heben sich klar vom Fließtext ab und sind hier aufgelistet:

Algorithmus 1 Beispiel Algorithmus

Diese Darstellung wird von Algorithmendarstellungen verwendet. Sie dienen zum Verständnis bestimmter Sachverhalte

Beispiel 1:
In diesen Boxen stehen Beispiele, die nicht notwendig sind um den Sachverhalt zu verstehen. Sie dienen lediglich zur Verdeutlichung von Algorithmen.

Hier kann zum Beispiel ein Quelltext stehen.

Abbildung 1.1: Beispielabbildung

Kapitel 2

Maschinelles Lernen

Tom Mitchell hat 1997 in seinem Buch „Machine Learning“ geschrieben:

> A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E.[Mit97]

Maschinelles Lernen bezeichnet also die computergestützte Erlangung von Wissen und Erfahrung. Ein System kann aus Beispielen lernen und diese für eine bestimmte Aufgabe generalisieren. Ein einfaches Speichern der Daten ist nicht ausreichend, weil ein neues noch unbekanntes Beispiel (in unserem Fall eine E-Mail) nicht erkannt werden würde. Somit ist die Generalisierung ein sehr wichtiger Aspekt des maschinellen Lernens.
Die Generalisierung kann vor dem Klassifizieren oder nach dem Klassifizieren geschehen. Alle hier gezeigten Verfahren benutzen bereits generalisierte Daten um die Daten zu klassifizieren. Die sogenannten „Lazy Learner“, die erst „auswendig“ lernen und beim Klassifizieren die Beispiele generalisieren, werden nicht weiter betrachtet. Eine Speicherung kompletter E-Mails ist aus Datenschutzgründen bedenklich und die Zeit für das Klassifizieren neuer E-Mails steigt erheblich.

Oft muss der „Lerner“ Kompromisse eingehen, um eine nicht zu große Spezialisierung zu erlangen. Das heißt, erkennt ein Klassifizierer alle Beispiele einer Trainingsmenge, ist es möglich, dass er sich sehr stark an die Trainingsmenge angepasst hat und eine Klassifizierung auf einer unabhängigen Testmenge sehr schlecht ausfällt, da diese Beispiele nicht in der Trainingsmenge vorkamen. Im Falle von E-Mail–Klassifikation werden die Worte getrennt und nicht die komplette E-Mail gespeichert.
Das Attachment-Erkennungsproblem lässt sich mit

- der Aufgabe T: Erkennen von anhangverdächtigen E-Mails anhand des Textes
- einem Performanzmaß P: Prozentsatz der korrekt klassifizierten E-Mails, wobei die Kosten für falsch klassifizierte E-Mails mit Attachment und E-Mails ohne Attachment gleich sind

- und der Trainingserfahrung E: Versendete E-Mails

beschreiben.
Um die Aufgabe T zu lösen, stelle ich im Verlauf der Arbeit folgende Algorithmen vor:

- Einfacher Algorithmus
- Erweiterter Algorithmus
- Naive Bayes Algorithmus
- Paul Graham Algorithmus
- Kombinierter Naive Bayes und Paul Graham Algorithmus (im weiteren Combined-Algorithmus genannt)

Das Performanzmaß gibt an, ob der Algorithmus eine gute Korrektklassifizierungsrate (=Accuracy) ermöglicht. Die Korrektklassifizierungsrate ist das Verhältnis zwischen den korrekt erkannten E-Mails zur Gesamtanzahl der E-Mails. Je besser das Maß, desto besser der eingesetzte Algorithmus. Besser bedeutet hier bei Accuracy möglichst eins.
Die Trainingserfahrung wird durch zwei Verfahren vergrößert. Um ein inkrementelles Lernen zu ermöglichen, wird jede E-Mail *M* nach dem Versenden an den Lerner geschickt. In Fachliteratur spricht man hier von „Online-Learnern“.

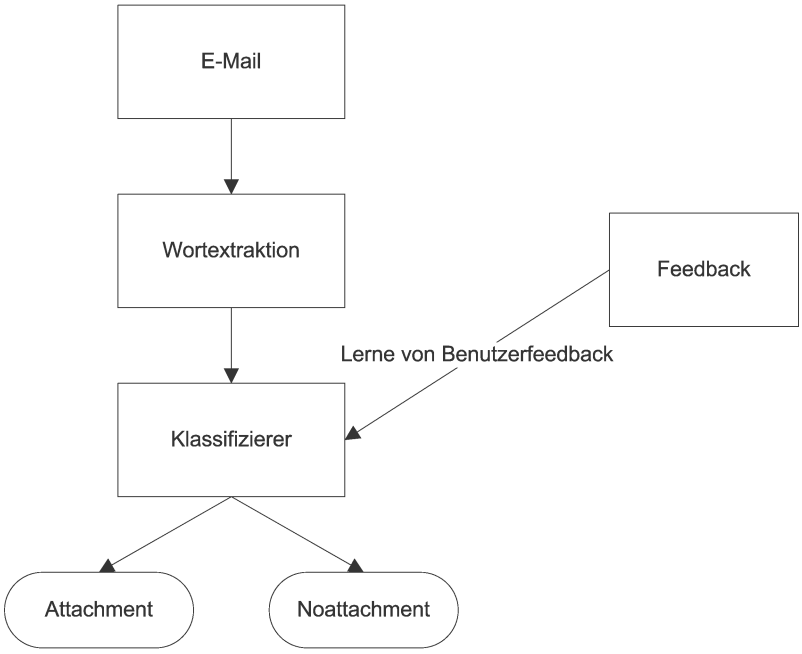

Abbildung 2.1: Schematische Darstellung des Ablaufs

Ein zweites Verfahren, dass eine Starterfahrung schafft, ist das Lernen bereits versendeter Nachrichten. Meist lernt man hier alle E-Mails, die im Ordner „Versendete Objekte“

(oder ähnlichen Ordnern) abgelegt sind. Abgebildet ist dieses Erlangen von Erfahrung mit einem Feedback-Kasten in der Abbildung 2.1.
Betrachtet man die Algorithmen, sieht man schnell, dass einige Klassifizierer die Erfahrung nicht sinnvoll verwenden.
Damit ergibt sich ein Ablauf, der zuerst die Worte aus den E-Mails („Wortextraktion“) extrahiert und anschließend eine Klassifizierung durchführt (vgl. Abbildung 2.1).

2.1 Wortextraktion

Jede E-Mail ist eine — nach Bedeutung — geordnete Menge an Worten. Da es sehr viele verschiedene Zeichen gibt, ist es sinnvoll, eine Menge an Zeichen zu definieren, die in einem Wort vorkommen dürfen. Bei diesen Zeichen sind meist die Buchstaben von A bis Z in Groß- und Kleinschreibung sowie der Bindestrich enthalten. Weitere Zeichen können je nach Anwendungsfall hinzukommen, sind aber beim Attachment-Problem nicht notwendig.
Ein anderer Weg wäre die Definition von Trennzeichen zwischen Worten. Dieser Weg wird hier nicht beschrieben, da dieser keine Vorteile gegenüber dem vorherigen bringt. Bei beiden Verfahren erhalten wir eine Wortmenge zur maschinellen Weiterverarbeitung. Algorithmus 2 zeigt, wie sich eine einfache Teilung durchführen lässt.

Algorithmus 2 einfache Wortextraktion

Vorbedingung: *TOKEN_RE*, *email* ist gegeben
Nachbedingung: Menge E_M mit einzelnen Worten
trenne *email* mit *TOKEN_RE* in E_M

Voraussetzung für den Algorithmus sind eine gesetzte Menge *TOKEN_RE*, sowie den E-Mail-Text *email*. *TOKEN_RE* gibt die Menge von Zeichen an, die innerhalb eines Wortes vorkommen dürfen. Als Ergebnis resultiert ebenfalls eine Wortmenge.
Da einige Algorithmen mit der Häufigkeit der Wörter eine Klassifizierung bestimmen, ist dies im Algorithmus 3 berücksichtigt. Dadurch beinhaltet die zurückgegebene Menge nicht nur die Worte, sondern auch deren Häufigkeit, die im Algorithmus durch die Variable *counter* dargestellt wird.

Die Variablen *emailcounter* und *tokencounter* zählen jeweils die Anzahl der E-Mails bzw. die Anzahl der Worte in den E-Mails.

Algorithmus 3 erweiterte Wortextraktion

Vorbedingung: *TOKEN_RE*, *email* ist gegeben
Nachbedingung: Menge E_M mit einzelnen Worten und deren Häufigkeit

```
trenne email mit TOKEN_RE in tokens
E_M = ⊘
if tokens ≠ ⊘ then
  inkrementiere emailcounter
end if
for all t in tokens do
  inkrementiere tokencounter
  if t in E_M then
    inkrementiere counter vom Wort t in E_M
  else
    E_M = E_M ∪ t
    setze counter = 0 vom Wort t in E_M
  end if
end for
```

Beispiel Algorithmus 2:
Gegeben ist eine Zeichenkette *email* mit dem Text der aktuellen E-Mail. Außerdem noch ein regulärer Ausdruck *TOKEN_RE* der erlaubten Zeichen.
email=“Das ist ein E-Mail-Text“
TOKEN_RE=[A-Za-z-]
$\Rightarrow E_M = \{Das, ist, ein, E-Mail-Text\}$

Der reguläre Ausdruck *TOKEN_RE* sagt aus, dass alle Buchstaben A bis Z und a bis z, sowie der Bindestrich in Wörtern enthalten sein dürfen.

2.2 Einfacher Algorithmus

Der einfache Algorithmus ist ein Verfahren, das nicht lernt. Jede E-Mail wird gleich behandelt und die Klassifizierung wird immer auf die gleiche Art und Weise durchgeführt. Dies bedeutet, dass der Algorithmus keine Erfahrung sammelt und dadurch keine bessere Klassifikation durchführen kann. Der Algorithmus bestimmt anhand der Wortexistenz in einer E-Mail, ob die E-Mail anhangverdächtig ist.
Für den einfachen Algorithmus wird die E-Mail M, bestehend aus den Worten t_i, mit dem einfachen Algorithmus zur Wortextraktion in die Menge E_M zerlegt. Eine weitere Menge W enthält frei wählbare Worte w_j, die im Vorfeld vom Benutzer definiert werden müssen.

$$E_M = \{t_1, t_2, t_3 ... t_n\} \text{ und } W = \{w_1, w_2, ... w_m\} \tag{2.1}$$

Gilt nun $E_M \cap W \neq \oslash$, dann ist die E-Mail M anhangverdächtig.
Algorithmus 4 verdeutlicht die Vorgehensweise dieses Verfahrens.

Algorithmus 4 Einfacher Algorithmus

```
Vorbedingung: E_M gesetzt
  for all t in E_M do
    if t in W then
      E_M ist anhangverdächtig
    end if
  end for
```

Beispiel:
Gegeben ist:

$$E_M = \{Das, ist, eine, E-Mail, mit, Anhang\}$$

$$W = \{Anhang, Anlage, anbei\}$$

dann gilt also $E_M \cap W = \{Anhang\} \neq \oslash$, somit ist diese E-Mail anhangverdächtig.
Daraufhin kann nun eine Aktion ausgelöst werden, die dem Nutzer einen Hinweis auf das Fehlen eines Attachments anzeigt.

Die Vorteile des Algorithmus sind die Einfachheit der Umsetzung, die Geschwindigkeit und der Speicherverbrauch. Da der Algorithmus nicht lernt, sammelt er keine Daten und somit bleibt der verbrauchte Speicher konstant niedrig. Die Geschwindigkeit hängt nur von der Größe der E-Mail, die klassifiziert werden soll, ab und ist somit sehr hoch. Es sind keine Berechnungen für eine Klassifikation notwendig.
Der größte Nachteil dieses Verfahrens ist, dass der Benutzer mit seiner Wortwahl W bestimmt, wie gut oder wie schlecht die Klassifikation abläuft. Wählt der Benutzer Worte, die nur sehr selten oder in der falschen Klasse von E-Mails vorkommen, wird das Ergebnis dieses Klassifizierers sehr schlecht sein. Ein Ansatz um das Problem der Wortwahl zu lösen stellt der erweiterte Algorithmus dar.

2.3 Erweiterter Algorithmus

Beim erweiterten Algorithmus versuche ich, das Problem der Wortwahl zu adressieren. Die Klassifikation einer E-Mail wird, wie beim einfachen Algorithmus, über die Wortexistenz gemacht (vgl. Algorithmus 4). Im erweiterten Algorithmus wird aber zur Wortextraktion der Algorithmus 3 genutzt.
Der Fokus liegt hier auf der automatischen Generierung der Wortmenge W. Eine Startmenge W_0 kann vom Benutzer frei gewählt werden. Die Qualität von W_0 bestimmt maßgeblich die Güte des Algorithmus bei kleinen E-Mail-Mengen, da der Algorithmus noch kein W bestimmt hat. Eine hohe Qualität bedeutet eine Startmenge zu wählen, die möglichst bei jeder Attachment-E-Mail-Menge ein zutreffendes Wort beinhaltet.

Die Generierung der Menge W geschieht über die Worthäufigkeiten. Um die Worthäufigkeiten in den E-Mails für jede Klasse zu bestimmen, teilt sich unsere Menge der E-Mails E in die beiden Mengen $E_{attachment}$ und $E_{noattachment}$

$$E = E_{noattachment} \cup E_{attachment} \tag{2.2}$$

In den Mengen $E_{attachment}$ und $E_{noattachment}$ werden alle Worte in Tupeln der Form $(token, count)$, wobei *token* das Wort und *count* ganzzahlig ist und die Anzahl von *token* in $E_{attachment}$ bzw. $E_{noattachment}$ angibt, erfasst.
Die Idee ist, die Worte beim Lernen einer E-Mail zu zählen und aufgrund eines Verhältnisses Wörter in die Menge W aufzunehmen oder zu entfernen.
Sei $x = (x_t, x_i)$ ein Tupel aus $E_{attachment}$ und $y = (y_t, y_i)$ ein Tupel aus $E_{noattachment}$, dann ist x_t das Wort und x_i die Anzahl. Gilt $x_t = y_t$ und $x_i > v \cdot y_i$ dann wird das Wort x_t in W aufgenommen. Der Faktor v kann frei gewählt werden und bestimmt ab welcher Häufigkeit das Wort berücksichtigt wird.
Gilt allerdings $x_t = y_t$ und $x_i < y_i$ wird das Wort x_t aus W entfernt und damit nicht mehr zur Klassifikation genutzt.

Algorithmus 5 Erweiterter Algorithmus - Lernprozess

Vorbedingung: att und E_M sind gegeben
  for all t in E_M **do**
    if att **then**
      inkrementiere t in $E_{attachment}$
    else
      inkrementiere t in $E_{noattachment}$
    end if
    if $x_t = y_t$ wobei x_e und $x_i > v \cdot y_i$ **then**
      $W = W \cup t$
    end if
    if $x_t = y_t$ wobei x_e und $x_i < y_i$ **then**
      $W = W \backslash t$
    end if
  end for

Im Lernprozess des Algorithmus werden alle Worte der E-Mail durchlaufen und in der jeweiligen Menge $E_{attachment}$ bzw. $E_{noattachment}$ eingefügt und dementsprechend die Anzahl der Worte erhöht. Die Variable $att \in \{true, false\}$ gibt an, ob die E-Mail M ein Attachment hatte oder nicht.

Beispiel:
Gegeben seien:

$$W = W_0 = \{anhang\}$$

$$E_{attachment} = \{(anhang, 100), (anlage, 100)\}$$

$$E_{noattachment} = \{(anhang, 1), (anlage, 10)\}$$

$$v = 8$$

Da $(anlage, 100) > v \cdot (anlage, 10) \rightarrow W \cup anlage = \{anhang, anlage\}$

Diese Erweiterung des einfachen Algorithmus beachtet die Worthäufigkeiten in den gelernten E-Mails. Da aber Wörter in Attachment-E-Mails und in E-Mails ohne Attachments vorkommen können, werden Fehlalarme ausgelöst.
Wenn v, also das Verhältnis zwischen Attachment-Wort und Noattachment-Wort, zu groß wird, dann werden keine Worte in W hinzugefügt, aber entfernt. Dadurch kann es passieren, dass $W = \oslash$ ist und keine Klassifikation mehr stattfindet.
In diesem Algorithmus wird auch nur auf Wortexistenz geprüft, dadurch werden andere Worte in den E-Mails vernachlässigt. Um alle Wörter einer E-Mail zu beachten, verwenden wir die beiden nachfolgenden Verfahren.

2.4 Naive Bayes

Naive Bayes Klassifizierer sind für ihre Einfachheit und die hohe Lerngeschwindigkeit bekannt. Der Begriff Naive rührt daher, dass angenommen wird, dass die Worte untereinander unabhängig sind. Diese Annahme ist natürlich nicht korrekt, da ein Benutzer Worte nur so wählt, dass sie Sinn machen, sonst würden E-Mails vom Empfänger nicht verstanden werden.[Cha03][Mit97]

Die E-Mail-Klassen „Attachment“ und „Noattachment“ werden nun c_1, c_2 genannt, hierbei entspricht die Klasse c_1 der Attachmentklasse und die Klasse c_2 der Noattachmentklasse.
Mit Hilfe des Satzes von Bayes, kann die Wahrscheinlichkeit, dass eine Klasse c_i auftritt unter der Bedingung, dass eine E-Mail M aufgetreten ist, bestimmt werden. Wir nennen diese

$$P(c_i|M) = \frac{P(M|c_i) \cdot P(c_i)}{P(M)}, \tag{2.3}$$

wobei

- $P(c_i)$ die A-Priori-Wahrscheinlichkeit für die Klasse c_i
  $\rightarrow$ Wahrscheinlichkeit, dass c_i eingetreten ist ohne Beachtung von M
- $P(M)$ die A-Priori-Wahrscheinlichkeit für die E-Mail M
  $\rightarrow$ Wahrscheinlichkeit, dass M eingetreten ist, ohne Beachtung von c_i

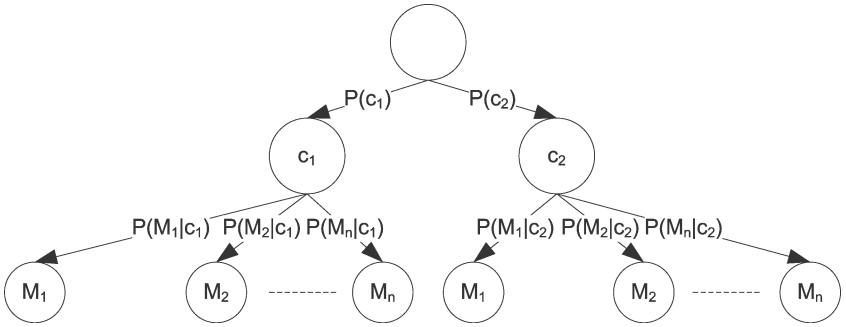

Abbildung 2.2: Wahrscheinlichkeitsbaum Bayes

- $P(M|c_i)$ die Wahrscheinlichkeit für eine E-Mail M unter der Bedingung, dass die Klasse c_i eingetreten ist.

Bildlich betrachtet entspricht $P(M)$ der Summe der Pfade, die zur E-Mail M führen (vgl. Abbildung 2.2). Mathematisch ausgedrückt bedeutet dies

$$P(M) = \sum_{c} P(M|c_i) \cdot P(c_i). \tag{2.4}$$

$P(M)$ kommt in der Formel 2.3 als Nenner vor und kann vernachlässigt werden, wenn nur ein Ranking und keine Wahrscheinlichkeiten benötigt werden. Daraus folgt direkt der Bayes Klassifizierer

$$c = argmax_c P(M|c_i) \cdot P(c_i). \tag{2.5}$$

c gibt an, welche Klasse c_i den höchsten Wahrscheinlichkeitswert hat.
Durch die Wortextraktion werden die E-Mails in Worte t_j geteilt und nicht als komplette Nachrichten — wie bisher angenommen — gespeichert (vgl. Abbildung 2.3). Die Annahme, dass Worte unabhängig voneinander sind, ermöglicht die Vereinfachung

$$P(M|c_i) = \prod_{j=1}^{|M|} P(t_j|c_i). \tag{2.6}$$

Mit $P(t_j|c_i)$ bezeichnet man nun die Wahrscheinlichkeit, dass ein Wort t_j in der Klasse c_i auftritt. Setzt man dieses ein, erhält man den Naive Bayes Klassifizierer

$$c = argmax_c \prod_{j=1}^{|M|} P(t_j|c_i) \cdot P(c_i). \tag{2.7}$$

Für den Naive Bayes Klassifizierer sind nun einzelne Worte t_j nötig. Die Menge der Worte E_M erhält man aus der E-Mail M mit dem erweiterten Algorithmus zur Wortextraktion (vgl. Algorithmus 3). Der Algorithmus 6 zum Lernen von Beispielen veranschaulicht, dass E-Mails mit Attachment in die Menge $E_{attachment}$ und E-Mails ohne Attachment in

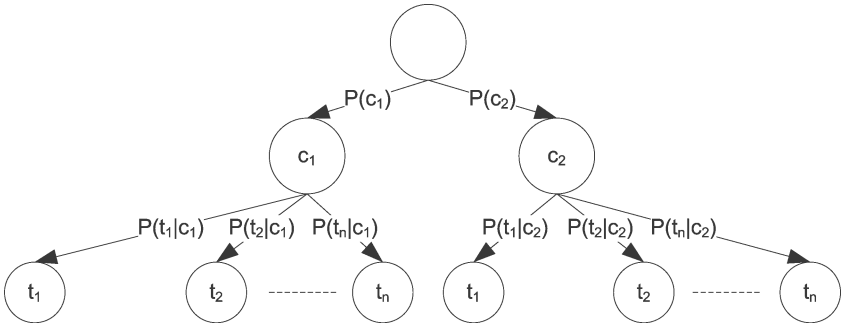

Abbildung 2.3: Wahrscheinlichkeitsbaum Naive Bayes

Algorithmus 6 Naive Bayes Lerner

Vorbedingung: E_M und *att* sind gegeben.
  for all t in E_M **do**
    if att **then**
      inkrementiere *counter* von t in $E_{attachment}$
    else
      inkrementiere *counter* von t in $E_{noattachment}$
    end if
  end for
  berechne alle $P(t_j|c_i)$ neu

die Menge $E_{noattachment}$ kommen. In beiden Mengen sind ebenfalls die Häufigkeiten der Worte enthalten. Die Variable $att \in \{true, false\}$ gibt an, ob die E-Mail ein Attachment enthält oder nicht. Die Berechnung der

$$P(t_j|c_i) = \frac{\text{Häufigkeit des Wortes } t_j \text{ in der Klasse } c_i}{\text{Gesamtanzahl aller Worte in der Klasse } c_i} \tag{2.8}$$

kann mit Hilfe einer Schleife über alle Worte realisiert werden. Man erhält zwei neue Mengen für Attachment und Noattachment mit den Wahrscheinlichkeitswerten. Die Einträge dieser beiden Mengen sind Tupel der Form $(token, p(token|c_i)$, wobei *token* das Wort ist und $p(token|c_i)$ die Wahrscheinlichkeit. Diese Vorberechnung dient zur Vereinfachung des Algorithmus.
Jetzt sind alle Wörter erfasst und deren Auftrittswahrscheinlichkeit berechnet, deswegen kann die Klassifizierung einer neuen E-Mail anhand des Algorithmus 7 durchgeführt werden.
Benötigt wird eine Menge an E-Mail-Worten E_M, die aus der neuen E-Mail M mit dem einfachen Wortextraktions-Algorithmus entsteht. Die Variablen *attp* und *noattp* geben jeweils die Wahrscheinlichkeit an, dass E_M zur Attachment- oder Nichtattachmentklasse gehört. Da es sich nur um ein Zweiklassenproblem handelt, werden beide Wahrscheinlichkeiten parallel berechnet. Hier wird angenommen, dass es ausreichend ist, wenn $attp > noattp$ ist, damit die E-Mail der Attachmentklasse zugeordnet wird.

Algorithmus 7 Naive Bayes Klassifizierer

Vorbedingung: E_M ist gegeben.
  setze $attp = P(c_1)$
  setze $noattp = P(c_2)$
  for all t in E_M **do**
    berechne $attp = attp \cdot P(t|c_1)$
    berechne $noattp = noattp \cdot P(t|c_2)$
  end for
  if $attp > noattp$ **then**
    E_M ist anhangverdächtig
  end if

Beispiel:
Gegeben sind folgende Wahrscheinlichkeiten:
email="Hier ist die Datei"
$E_{attachment} = \{(datei, 2), (hier, 20), (ist, 10), (hallo, 10), (die, 10)\}$
$E_{noattachment} = \{(hier, 3), (ist, 10), (hallo, 10), (die, 10), (und, 15)\}$
$|E| = |E_{attachment}| + |E_{noattachment}| = 52 + 48 = 100$

$$attp = \frac{52}{100} \cdot \frac{20}{52} \cdot \frac{10}{52} \cdot \frac{10}{52} \cdot \frac{2}{52} = 0.0002845 \rightarrow 95.63\%$$

$$noattp = \frac{48}{100} \cdot \frac{3}{48} \cdot \frac{10}{48} \cdot \frac{10}{48} \cdot \frac{1}{100} = 0.000013 \rightarrow 4.36\%$$

Nach der Normierung ist klar, dass diese E-Mail anhangverdächtig ist.
Alle Werte hier sind rein fiktiv und sind niemals in dieser Zusammensetzung in E-Mails vorgekommen.

2.5 Paul Grahams Algorithmus

Paul Graham ist ein Essayist, Programmierer und Programmiersprachendesigner. 2002 hat er in seinem Paper „A Plan for a Spam" einen einfachen statistischen Spam-Filter beschrieben, der von vielen aktuellen Spam-Filtern in Ansätzen übernommen wurde. [Gra02]
Der Algorithmus von Paul Graham — im weiteren APS genannt — ist für das Spam Problem, ähnlich wie der Naive Bayes, gut geeignet. Für das Attachmentproblem sind hier kleine Anpassungen nötig. Diese Anpassungen beziehen sich hauptsächlich auf die Gewichtung der beiden Klassen Attachment und Noattachment.

Wie auch die beiden vorherigen Algorithmen arbeitet APS auf den Wortlisten $E_{attachment}$ und $E_{noattachment}$. Diese entstehen durch den Algorithmus 8.
Dieser Algorithmus benötigt die Menge E_M, die durch den erweiterten Wortextraktions-

Algorithmus 8 APS - Tokenize

Vorbedingung: E_M und *att* sind gegeben.
  for all t in E_M **do**
    if att **then**
      inkrementiere *counter* von t in $E_{attachment}$
    else
      inkrementiere *counter* von t in $E_{noattachment}$
    end if
  end for
  Rufe Vorberechnung auf (Algorithmus 9).

Algorithmus entsteht.
Nach Erstellen der Mengen $E_{attachment}$ und $E_{noattachment}$ wird die Vorberechnung der Werte m_{noatt} und m_{att} durchgeführt. Diese geben eine Kennzahl für die Häufigkeit der Worte in den verschiedenen E-Mail Mengen an. Damit diese Werte im Bereich zwischen 0.0 und 1.0 bleiben, wird in

$$m_{noatt} = min(1.0; (\frac{NF \cdot n_{t,noattachment}}{|E_{noattachment}|})) \tag{2.9}$$

und

$$m_{att} = min(1.0; (\frac{AF \cdot n_{t,attachment}}{|E_{attachment}|})) \tag{2.10}$$

das Minimum zwischen 1.0 und dem Verhältnis zwischen der Anzahl der Worte in den Mengen und der Anzahl der E-Mails berechnet. $n_{t,noattachment}$ und $n_{t,attachment}$ bezeichnen jeweils die Häufigkeit des Wortes t in der Menge $E_{noattachment}$ bzw. $E_{attachment}$. $|E_{attachment}|$ und $|E_{noattachment}|$ folglich die Anzahl der gelernten E-Mails in den beiden Mengen. Im Spam-Problem hat Paul Graham $NF = 1$ und $AF = 2$ gewählt, in unserem Problem ist aber $NF = AF = 1$ besser geeignet und damit sind beide Klassen gleichgewichtet.
Daraus ergibt sich eine Gesamtmetrik m_{total}, die ein Indikator dafür ist, wie oft ein Wort t in E-Mails mit bzw. ohne Attachments aufgetreten ist. Umso größer der Wert ist, desto höher ist die Auftrittswahrscheinlichkeit des Wortes in der Attachment-Klasse.
Betrachtet man

$$m_{total} = max(NP, min(AP, \frac{m_{att}}{m_{att} + m_{noatt}})), \tag{2.11}$$

wird wieder ein Wert zwischen 0 und 1 angenommen. NP gibt den niedrigsten Wert an, den ein Wort haben kann, wenn es in der Noattachment-Klasse enthalten ist. Analog gibt AP den höchsten Wert an, den ein Wort haben kann, wenn es in der Attachment-Klasse enthalten ist. Da $0 < NP < 1$, $0 < AP < 1$ und $NP < AP$ gilt, wird der Wert von m_{total} einen Wert zwischen 0 und 1 annehmen.
Wird dies für alle Wörter aus $E_{attachment} \cup E_{noattachment}$ wiederholt, ergibt sich eine Menge *db* in der Tupel $(token, m_{total})$ gespeichert sind. (vgl. Algorithmus 9)

Algorithmus 9 APS - Lernen

Vorbedingung: $E_{attachment} \neq \oslash \vee E_{noattachment} \neq \oslash$
Nachbedingung: db hat m_{total} für alle Wörter aus $E_{attachment} \cup E_{noattachment}$
  for all t in $E_{attachment} \cup E_{noattachment}$ **do**
    $g = NF \cdot$ Häufigkeit von t in $E_{noattachment}$
    $b = AF \cdot$Häufigkeit von t in $E_{attachment}$
    if $g + b \geq FREQUENCY\_THRESHOLD$ **then**
      $m_{noatt} = min(1.0, \frac{g}{|E_{noattachment}|})$
      $m_{att} = min(1.0, \frac{b}{|E_{attachment}|})$
      $m_{total} = max(NP, min(AP, \frac{m_{att}}{m_{att}+m_{noatt}}))$
      setze m_{total} von t
    end if
  end for

FREQUENCY_THRESHOLD ist ein Integer, der angibt wie oft ein Wort t vorkommen muss, bevor es in die Menge db aufgenommen wird. Diese Berechnungen werden nur dann ausgeführt, wenn ein neues Wort gelernt wird.

Algorithmus 10 zeigt, wie die Klassifizierung einer neuen E-Mail durchgeführt wird.

Algorithmus 10 APS - Klassifizieren

Vorbedingung: $E_M \neq \oslash$
  $pairs = \oslash$
  for all t in E_M **do**
    $pairs \cup (t, m_{total,t})$
  end for
  Sortiere $pairs$ mit $m_{total} - 0.5$
  $significant = \{x_1, ..., x_{15}\} \in pairs$
  $inverseProduct = Product = 1.0$
  for all t in $significant$ **do**
    $product = product * m_{total,t}$
    $inverseProduct = inverseProduct = 1.0 - prob$ of t
    return $\frac{product}{product+inverseProduct}$
  end for

Damit ist die Klassifizierung durch

$$m_{final} = \frac{\prod_{i=1}^{15} m_{total,i}}{\prod_{i=1}^{15} m_{total,i} + \prod_{i=1}^{15}(1 - m_{total,i})} \tag{2.12}$$

gegeben. $m_{total,i}$ gibt den m_{total}-Wert des i-ten Wortes der signifikanten 15 Wörter an. Die signifikanten Wörter ergeben sich aus den Wörtern mit den höchsten m_{total}-Werten in db.

Da hier kein Vergleichswert mit der „Noattachment“-Klasse vorliegt, muss der Schwellwert, ab dem eine E-Mail anhangverdächtig ist, gewählt werden.

2.6 Combined-Algorithmus

Da die beiden Algorithmen Naive Bayes und Paul Graham auf unterschiedliche Art und Weise die Beispiele klassifizieren, ist es möglich, eine Kombination zu bilden. Diese Kombination ist durch das Hintereinanderausführen der Algorithmen möglich.
Zuerst wird Naive Bayes auf die E-Mail angewendet, danach Paul Graham. Beide geben Werte zurück, die angeben, wie hoch die Wahrscheinlichkeit einer anhangverdächtigen E-Mail ist. Die Ergebnisse der beiden Algorithmen können durchaus verschieden ausfallen, dadurch ist eine Kombination überhaupt erst sinnvoll.

$$p_{total} = \frac{v \cdot NB + APS}{v + 1}, \tag{2.13}$$

wobei

- v Faktor, wie hoch NB gewichtet wird,
- NB ist der Wahrscheinlichkeitswert vom Naive Bayes für die Attachment-Klasse,
- APS ist der Wahrscheinlichkeitswert von APS-Algorithmus für die Attachment-Klasse,

gibt nun einen kombinierten Wert zurück. Der Wert $v = 2$ wurde experimentell ermittelt und wird als Grundlage für die Evaluierung verwendet.

Beispiel:
$NB = 0.78$
$APS = 0.72$
$v = 2$

$$p_{total} = \frac{2 \cdot 0.72 + 0.78}{2 + 1} = 0.74$$

In diesem Beispiel ist die Wahrscheinlichkeit hoch, dass die E-Mail anhangverdächtig ist.

Wie aus dem Beispiel schon zu erkennen, muss man bei diesem Algorithmus einen Schwellwert festlegen, ab wann eine E-Mail als anhangverdächtig gilt.

Kapitel 3

Implementierung

Der AttachmentChecker besteht aus zwei Komponenten: dem Thunderbird-Plugin und dem Server (vgl. Abbildung 3.1).

Abbildung 3.1: Schematischer Aufbau

Der Server ist in Python realisiert und enthält alle Algorithmen aus Kapitel 2. Das Thunderbird-Plugin – im weiteren Plugin genannt – ist komplett in JavaScript geschrieben. Diese Aufteilung wurde gewählt, weil in JavaScript das Testen der Algorithmen erschwert würde. Für JavaScript gibt es im Gegensatz zu Python keinen eigenständigen Interpreter, was eine Evaluierung ohne Thunderbird nicht unwesentlich erschweren würde (siehe Abschnitt 3.3).
Da Server und Plugin getrennt voneinander wirken, muss eine Kommunikation zwischen beiden hergestellt werden. Sie kommunizieren über eine Socket-Verbindung per XML miteinander.
Das Plugin benötigt somit Funktionen zum

- Starten des Servers,
- Prüfen, ob der Server bereits läuft,
- Beenden des Servers,
- Senden von Trainingsbeispielen und
- zum Senden für Klassifizierungsanfragen.

Der Server wiederum verrichtet die Arbeit der

- Klassifizierung,
- des Lernens und
- des Sendens und Empfangens von Beispielen.

Damit beim Nutzer der Eindruck gewahrt bleibt, er würde nur ein einziges Programm nutzen, wird der Server innerhalb von Thunderbird aufgerufen und auch wieder beendet. So ist es möglich, den in Python geschriebenen Server „unsichtbar“ zu benutzen.

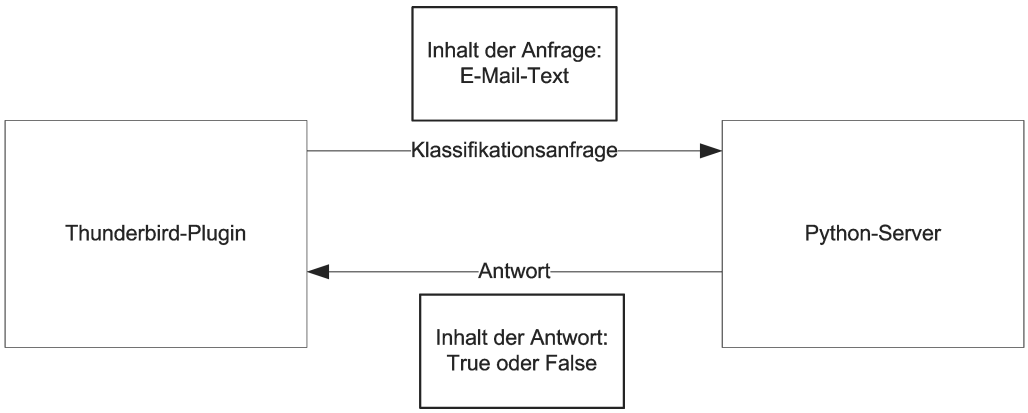

Abbildung 3.2: Schematischer Ablauf - Klassifikationsanfrage

Der Ablauf einer Klassifikation ist in Abbildung 3.2 veranschaulicht. Thunderbird schickt den E-Mail-Text an den Server und dieser prüft mit einem Algorithmus, ob es sich um eine anhangverdächtige E-Mail handelt. Meldet er „True“ zurück, ist sie anhangverdächtig, andernfalls sendet er „False“. Wenn die E-Mail anhangverdächtig ist, prüft das Plugin, ob vom Benutzer ein Attachment angehängt wurde. Ist dies der Fall, wird die E-Mail versendet. Ansonsten wird ein Popup angezeigt, dass dem Benutzer mitteilt, dass die E-Mail anhangverdächtig ist und ihm die Möglichkeit gibt ein Attachment anzuhängen. Ist die E-Mail nicht anhangverdächtig wird kein Popup angezeigt.
Jede E-Mail wird nach dem Sendevorgang nochmals an den Server geschickt um sie dort zu lernen. (vgl. Abbildung 3.3)

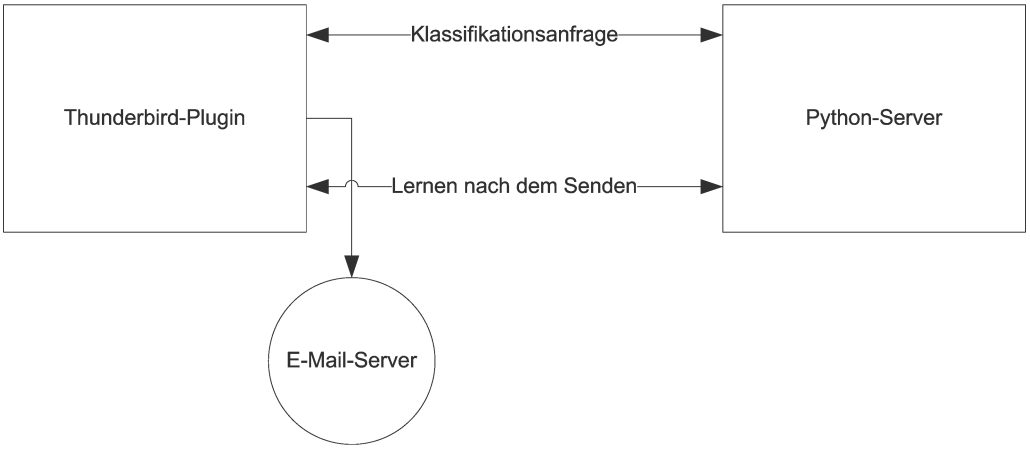

Abbildung 3.3: Schematischer Ablauf - Senden

3.1 Thunderbird

Mozilla Thunderbird ist ein sehr bekanntes E-Mail-Programm, das für Entwickler die Möglichkeit bietet, Erweiterungen mit Hilfe von Plugins einzubinden. [Moz08c]
Für die Erstellung eines Plugins benötigt man Kenntnisse in JavaScript, XUL und im allgemeinen Aufbau solcher Plugins.[Moz08b]

3.1.1 JavaScript und XUL

JavaScript ist eine Skriptsprache, die hauptsächlich in Web-Browsern eingesetzt wird, um dynamische Webseiten zu generieren.
In Thunderbird werden Plugins ausschließlich in JavaScript geschrieben. Kombinationen von XPCOM und JavaScript machen es möglich auch andere Sprachen zu verwenden.
XPCOM steht für Cross Platform Component Object Model und ermöglicht eine Reflektion von fremden Sprachelementen auf JavaScript. Dies bedeutet, dass C++-Komponenten mit einem JavaScript-Befehl aufrufbar sind. Viele Funktionen von Thunderbird sind so auch per JavaScript nutzbar. [Wik07]
Der AttachmentChecker hat folgende vier JavaScript-Dateien:

- server.js - Steuerung des Python-Servers
- attcheck.js - Funktionen zum Beeinflussen des Sendevorgangs
- attcheckContext.js - Funktionen zum Lernen über das Kontextmenü
- settings.js - Funktionen des Einstellungsdialogs

XUL steht für XML User-Interface Language. Dies ist eine plattformübergreifene Sprache für das Beschreiben von Benutzeroberflächen darstellt. [Moz07]

In Thunderbird wird XUL für die Definition der graphischen Benutzeroberfläche und für die Internationalisierung der Oberfläche verwendet.
Im AttachmentChecker gibt es fünf XUL-Definitionsdateien:

- attcheck.xul - Definition, welche der Sprachdateien im „E-Mail schreiben“-Fenster verwendet wird (vgl. Abbildung 3.4),
- attcheck_lang.xul - Definition, welche der Sprachdateien verwendet wird,
- attcheckContextOverlay.xul - Anzeige des Trainingseintrags im Kontextmenü,
- attcheckTrainOverlay.xul - Definition der Oberfläche beim Lernen von E-Mails,
- settings.xul - Definition der Oberfläche für die Einstellungen

```
1 <?xml version="1.0"?>
2
3 <overlay xmlns="http://www.mozilla.org/keymaster/gatekeeper/
    there.is.only.xul">
4   <script type="application/x-javascript" src="chrome://global/
      content/nsUserSettings.js"/>
5   <script type="application/x-javascript" src="chrome://
      attcheck/content/server.js"/>
6   <script type="application/x-javascript" src="chrome://
      attcheck/content/attcheck.js"/>
7
8   <stringbundleset id="stringbundleset">
9         <stringbundle id="LocalBundle" src="chrome://attcheck/
            locale/attcheck_lang.properties"/>
10  </stringbundleset>
11 </overlay>
```

Abbildung 3.4: Beispiel attcheck.xul

3.1.2 Allgemeiner Aufbau eines Thunderbird-Plugins

Ein fertiges Plugin wird in einer XPI-Datei gespeichert. Diese Datei ist im Prinzip lediglich ein komprimiertes ZIP-Archiv. Bis auf die Dateiendung müssen keine Änderungen am Dateiformat vorgenommen werden.
Der Inhalt der Datei sollte — wie in Abbildung 3.5 zu sehen — aufgebaut sein. [Moz08a]

```
/install.rdf
/components/*
/components/cmdline.js
/defaults
/defaults/preferences/*.js
/plugins/*
/chrome.manifest
/chrome/icons/default/*
/chrome/
/chrome/content/
```

Abbildung 3.5: Plugin Basis-Verzeichnisstruktur

Die Dateien `install.rdf` und `chrome.manifest` umfassen alle Informationen, die Thunderbird benötigt, um das Plugin ordnungsgemäß zu installieren und einzubinden. In der Datei `install.rdf` wird die ID, der Name, die Version, eine kurze Beschreibung, der Ersteller, sowie Informationen zu unterstützen Versionen angegeben. `chrome.manifest` beinhaltet alle Verzeichnisangaben und Sprachangaben.
Im Verzeichnis `/chrome` und dessen Unterordnern befinden sich die Daten des Plugin.
In den meisten Fällen verwendet man allerdings eine leicht abgewandelte Form, da `/compontents/*` oder `/components/cmdline.js` nur dann Verwendung finden, wenn eigene XPCOM-Komponenten benötigt werden. `/defaults` kann auch innerhalb des Verzeichnises `chrome` angelegt werden. Damit ergibt sich eine angepasste Struktur (vgl. Abbildung 3.6).

```
/install.rdf
/chrome.manifest
/chrome/
/chrome/content/
/chrome/content/locales
/chrome/content/defaults
```

Abbildung 3.6: Plugin Angepasste Verzeichnisstruktur

3.1.3 Aufbau des Plugins

Die Struktur aus Abbildung 3.6 ist die Grundlage für das AttachmentChecker-Plugin. Jedes Plugin hat eine eindeutige ID, damit es zu keinen Kollisionen innerhalb der Plugins kommen kann. Der AttachmentChecker hat die ID „{303160d3-d4b2-4ad4-b793-15d8a3969025}“, damit ist das Plugin eindeutig auffindbar. Existieren zwei Plugins mit der gleichen ID ist der fehlerfreie Betrieb nicht mehr garantiert.
Der Name „AttachmentChecker“ steht in keinem Zusammenhang mit der ID und kann somit frei gewählt werden. Dieser wird nach der Installation im Add-ons-Manager von Thunderbird angezeigt. Des Weiteren kann noch eine Beschreibung angeben werden, die direkt unter dem Namen des Plugins im Add-ons-Manager (Abbildung 3.7) erscheint.

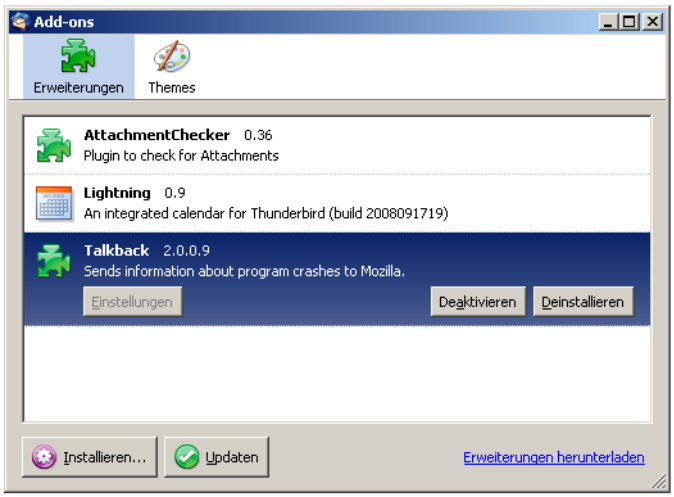

Abbildung 3.7: Thunderbird Add-ons-Manager

Die oben genannten Werte sind in der Datei `install.rdf` zu finden. Diese Datei beinhaltet ferner die Versionsnummer des Plugins, die Angabe, ob der Einstellungsdialog vorhanden ist und ob der Updateservice genutzt wird und für welche Zielplattform das Plugin konzipiert wurde (vgl. Abbildung 3.8). Die Zielplattform-ID beim AttachmentChecker ist „{3550f703-e582-4d05-9a08-453d09bdfdc6}“ und entspricht damit der ID von Thunderbird.

```
...
  <em:id>{303160d3-d4b2-4ad4-b793-15d8a3969025}</em:id>
  <em:name>Attachment Checker</em:name>
  <em:version>0.35</em:version>
  <em:description>Attachment Check before sending mail.</em:
      description>
  <em:creator>M. Ghiglieri</em:creator>
  <em:optionsURL>chrome://attcheck/content/settings.xul</em:
      optionsURL>
  <em:updateURL>http://www.mgnis.de/attachmentchecker/update.
      rdf</em:updateURL>

  <em:targetApplication>
   <Description>
    <em:id>{3550f703-e582-4d05-9a08-453d09bdfdc6}</em:id>
    <em:minVersion>1.0</em:minVersion>
    <em:maxVersion>2.0.0.*</em:maxVersion>
   </Description>
  </em:targetApplication>
...
```

Abbildung 3.8: Ausschnitt install.rdf

Eine weitere Konfigurationsdatei ist die Datei `chrome.manifest`. Dort werden die zu ladenden XUL-Dateien genauer definiert. Außerdem wird angegeben, wo sich die Sprachdateien befinden und zu welchem Ländercode sie gehören. Derzeit werden die Sprachen Deutsch (de) und Englisch (en) unterstützt (vgl. Abbildung 3.9).

```
content attcheck chrome/content/
overlay chrome://messenger/content/messengercompose/
    messengercompose.xul chrome://attcheck/content/attcheck.xul
overlay chrome://messenger/content/mailWindowOverlay.xul chrome
    ://attcheck/content/attcheckContextOverlay.xul
overlay chrome://messenger/content/messenger.xul  chrome://
    attcheck/content/attcheck_lang.xul
locale attcheck de chrome/locale/de/
locale attcheck en chrome/locale/en/
```

Abbildung 3.9: Datei chrome.manifest

Weitere Sprachen sind durch Hinzufügen des jeweiligen Ländercodes und einer geeigneten Sprachdatei möglich. Die Grundstruktur des Plugins ist damit geschaffen.

3.1.4 Funktionalität des Plugins

Die Aufgabe des Plugins ist das Senden und Empfangen von Daten und das Auslesen einzelner E-Mails. Einstellungen, die den AttachmentChecker betreffen, können über ein Einstellungsdialog vorgenommen werden. Die Internationalisierung bietet die Möglichkeit die Oberfläche in mehreren Sprachen darzustellen. Um immer auf dem neuesten Stand zu bleiben, findet der Updatemechanismus von Thunderbird Verwendung.

3.1.4.1 Datenversand und -empfang vom Server

Alle Funktionen, die benötigt werden, um die Verbindung zum Server herzustellen, befinden sich in der Datei `server.js`. Im Einzelnen sind das die Funktionen, um

- den Server zu starten (`launchServer`),
- den Server zu beenden (`shutdownServer`),
- zu prüfen, ob der Server gestartet ist (`checkServer`) und
- um Daten zum Server zu schicken und zu empfangen (`sendXMLMessage` und `sendwoXMLMessage`)).

In Klammern sind jeweils die betreffenden Funktionsnamen angegeben.

Der Server wird gestartet, indem Python mit der Datei `server.py` aufgerufen wird. Da Python im Plugin enthalten ist, muss festgestellt werden, wo sich das Plugin und damit Python befindet. Um Zugriff auf das Plugin-Verzeichnis zu bekommen wird ein XPCOM-Befehl `DIR.get` benutzt, der den Pfad zum Plugin-Verzeichnis wiedergibt (vgl. Abbildung 3.10).

```
1  function getChromeDir() {
2          var chromeDir = DIR.get("ProfD",Components.interfaces.
               nsIFile);
3          if (chromeDir.exists()) {
4                  chromeDir.append("extensions");
5                  chromeDir.append(ATTCHECKER_EXTENSION_ID);
6                  chromeDir.append("chrome");
7                  return chromeDir;
8          }
9          return null;
10 }
```

Abbildung 3.10: Funktion zum Auslesen des Verzeichnisnamens

Die Konstante ATTCHECKER_EXTENSIONS_ID ist festgelegt und muss den gleichen Inhalt, wie der ID-Wert in der Datei `install.rdf` haben. Diese Konstante muss gesetzt

sein, da es nicht möglich ist, diese direkt auszulesen. Sind alle Pfade bestimmt, kann der Server gestartet werden.
Unterschieden wird noch, ob es sich um das Betriebssystem Windows oder Linux handelt. Bei Linux kann das beigefügte Python nicht genutzt werden. Hier ist eine installierte Version notwendig. Das Plugin versucht Python dann in den Standard-Pfaden `/usr/bin/python` oder `/usr/local/bin/python` zu finden. Ist das Starten nicht möglich, wird eine Fehlermeldung ausgegeben (vgl. Abbildung 3.11).

Abbildung 3.11: Fehlermeldung, wenn Python nicht gefunden wurde

Sobald der Befehl zum Starten des Servers erfolgreich versendet wurde, wird fünf Sekunden gewartet. Diese Wartezeit ist notwendig, damit der Server ordnungsgemäß starten kann.
Die Kommunikation verläuft ab diesem Zeitpunkt über XML. Es wird eine Anfrage erstellt und diese wird an den Server geschickt. Der Server gibt die Daten aus der Anfrage an die jeweilige Funktion des Algorithmus und dieser erzeugt eine entsprechende Rückmeldung. Das Senden und Empfangen der Daten übernehmen die Funktionen `sendXMLMessage`, `sendwoXMLMessage` und `internalSendXMLMessage`. Die Funktion `internalSendXMLMessage` regelt die Übertragung zum Server.

```
1  function sendXMLMessage(data,type)
2  {
3          if (!checkServer()) { launchServer(); }
4          return internalSendXMLMessage(data,type);
5  }
6
7  function sendwoXMLMessage(data,type)
8  {
9          return internalSendXMLMessage(data,type);
10 }
```

Abbildung 3.12: Funktion zum Senden und Empfangen von Daten

Aufgerufen wird immer `sendXMLMessage` oder `sendwoXMLMessage`. Die beiden Funktionen unterscheiden sich dadurch, dass die erste Funktion vor jedem Befehl prüft, ob der Server bereits läuft. Diese Prüfung wird mit einer Serveranfrage, die in der Funktion `checkServer` implementiert ist, realisiert (vgl. Abbildung 3.13).

```
1 function checkServer()
2 {
3         response=internalSendXMLMessage("hello",'checkrun');
4         if (response=="OK") return true;
5         return false;
6 }
```

Abbildung 3.13: Funktion zum Prüfen, ob der Server bereits läuft

Dies meint, dass eine Anfrage erzeugt wird, indem eine Nachricht an den Server gesendet wird. Lautet die Serverantwort „OK“, darf der nächste Befehl ausgeführt werden. Kommt während einer bestimmten Zeit keine Antwort vom Server, wird von einem Timeout ausgegangen und der Server wird gestartet.
Mit Hilfe eines Listeners wird der Server beendet, sobald Thunderbird geschlossen wird. Ein Listener ist eine Funktion, die auslöst wird, wenn ein bestimmtes Ereignis eintritt, in diesem Fall das Schließen von Thunderbird.

3.1.4.2 Auslesen von E-Mails

Alle Algorithmen — bis auf den einfachen Algorithmus — klassifizieren besser, wenn sie die bereits gesendeten E-Mails kennen. Ein manuelles Lernen dieser E-Mails ist sinnvoll, wenn das Plugin installiert per Update aktualisiert wurde.
Dazu gibt es im Kontextmenü den Menüpunkt „Lerne AttachmentChecker“ (engl.: Train AttachmentChecker; Abbildung 3.14), der eine oder mehrere E-Mails aus der Übersicht an den Server schickt, damit sie dort gelernt werden.

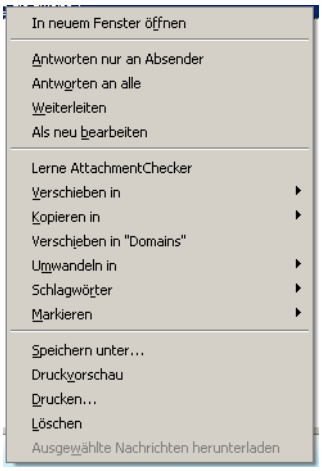

Abbildung 3.14: Kontextmenü AttachmentChecker

Thunderbird stellt keine Funktionen bereit, um nur den Text der E-Mail auszulesen. Man kann lediglich die komplette E-Mail mit den kompletten Attachments auslesen. Würde

man diese jedoch direkt an den Server schicken, wäre die Menge der übertragenen Daten sehr hoch.
Dieses Problem kann behoben werden indem man eine Trennung der E-Mails bereits per JavaScript in Thunderbird realisiert. Diese Trennung der E-Mails wird nach RFC 2822 durchgeführt. Dort wird beschrieben, wie E-Mails aufgebaut sein müssen, damit sie standardkonform sind. [rfc]
Eine E-Mail besteht aus zwei Teilen: Header und Body. Im Header sind unter anderem Informationen über die Herkunft, das Ziel und Informationen über die Trennung des Bodys enthalten. Diese Informationen über die Trennung des Bodys werden von E-Mail-Programmen genutzt, um den Text von den Attachments zu extrahieren. Somit ergibt sich eine Zusammensetzung des Bodys aus dem Text und den Attachments. Für den AttachmentChecker ist es belanglos, ob die E-Mail ein oder mehrere Attachments besitzt. Somit ist es also ausreichend, den Text und das erste Attachment zu bestimmen.
Die Trennung der einzelnen Bestandteile im Body wird mit dem Boundary, der im Header angegeben ist, durchgeführt. Der Boundary ist eine eindeutige Zeichenfolge, die in dieser Form in der E-Mail nicht auftritt. Ist der Boundary im Header nicht gegeben, wird davon ausgegangen, dass die E-Mail kein Attachment hat und der E-Mail-Text kann direkt extrahiert werden. Um zu verdeutlichen wie eine E-Mail aufgebaut ist, ist in Abbildung 3.15 skizziert, wie Body und Header aufgebaut sind, wenn die E-Mail zwei PDF-Dateien enthält.

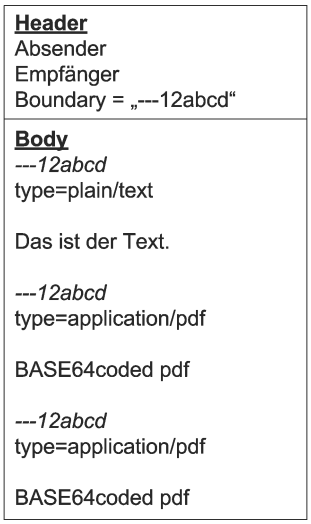

Abbildung 3.15: Aufbau E-Mail

Befindet sich im Header der Boundary, wird zuerst der E-Mail-Text gesucht, der vor den Attachments stehen muss, und danach das erste Attachment. Geachtet werden muss darauf, dass ein Text als Reintext und als HTML-Text enthalten sein kann.
Sind Text und Vorkommen von mindestens einem Attachment bestimmt, kann dem Server das Trainingsbeispiel geschickt werden.
Zusammengefasst gliedert sich der Ablauf in fünf Punkte:

1. Analysiere Header
2. Extrahiere die einzelnen Teile der E-Mail
3. Bestimme Text und Attachments
4. Sende E-Mail an Server
5. Sind noch weitere E-Mails ausgewählt, weiter bei 1.

3.1.4.3 Senden einer E-Mail

Die Aufgabe des AttachmentCheckers ist es, den Benutzer auf eine anhangverdächtige E-Mail aufmerksam zu machen. Das Plugin muss die ursprünglichen Funktionen zum Senden von E-Mails beeinflussen, um den Benutzer frühzeitig auf eine anhangverdächtige E-Mail hinweisen zu können.
Um Einfluss auf den Sendeprozess zu bekommen, müssen die Funktionen `SendMessage`, `SendMessageWithCheck` und `SendMessageLater` überschrieben werden.

```
1 function SendMessage() { if(SendExtension()) return; else
    GenericSendMessage(nsIMsgCompDeliverMode.Now); }
2 function SendMessageWithCheck() { if(SendExtension()) return;
    else GenericSendMessage(gIsOffline ? nsIMsgCompDeliverMode.
    Later : nsIMsgCompDeliverMode.Now); }
3 function SendMessageLater() { if(SendExtension()) return; else
    GenericSendMessage(nsIMsgCompDeliverMode.Later); }
```

Abbildung 3.16: Ausschnitt attcheck.js — Funktionen zum Senden und Empfangen von Daten

Die Abbildung 3.16 zeigt, dass bei jeder dieser Funktionen die ursprüngliche `GenericSendMessage`-Funktion aufgerufen wird, die das Senden der E-Mail übernimmt. Es werden hier drei Fälle behandelt: das direkte Senden, das spätere Senden und das Speichern als Entwurf. In allen drei Fällen wird die E-Mail vor der weiteren Verarbeitung geprüft.
Beim Prüfen einer E-Mail auf ein Attachment, wird nur der E-Mail-Text übertragen. Die Antwort wird ausgewertet und je nach Ergebnis ein Popup mit dem Hinweis, dass diese E-Mail möglicherweise ein Attachment enthalten sollte, angezeigt (vgl. Abbildung 3.17).
Jetzt besteht für den Benutzer die Möglichkeit ein Attachment anzuhängen oder die E-Mail ohne Attachment zu verschicken. Wird ein Attachment angehängt, wird der Sendeprozess ein weiteres Mal gestartet und die E-Mail direkt versandt.
Jede erfolgreich versandte E-Mail wird dem Server zum Lernen geschickt, um zukünftige E-Mails mit ähnlichen Wörtern besser einstufen zu können.

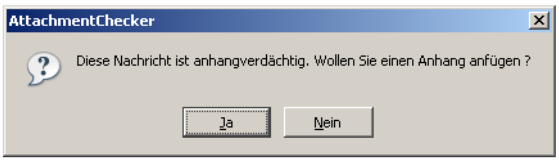

Abbildung 3.17: Popup, dass angezeigt wird, wenn die E-Mail anhangverdächtig ist

Durch diesen permanenten Lernprozess ist es möglich, dass der AttachmentChecker selbstständig ohne Einflussnahme des Benutzers lernen kann. Das Lernen einer Startmenge ist daher nicht notwendig, aber sehr zu empfehlen, da die Anzahl der Fehlklassifikationen dadurch stark abnimmt.

3.1.4.4 Einstellungsdialog

Der Einstellungsdialog bietet die Optionen, den AttachmentChecker und den Debugmodus ein- bzw. auszuschalten. Der Debugmodus steht für Testzwecke zur Verfügung und zeigt in einem gesonderten Fenster an, was der Server gerade macht. Im normalen Betrieb sollte der Debugmodus ausgeschaltet sein.
Es ist möglich den Port des Python-Servers zu beeinflussen. Der Standard-Port ist 9999.

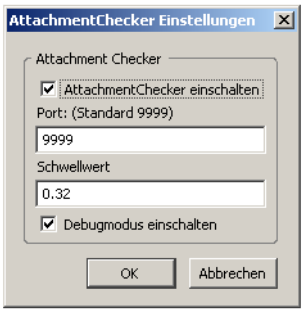

Abbildung 3.18: Einstellungsdialog AttachmentChecker

Des Weiteren ist der Schwellwert, ab dem eine E-Mail als anhangverdächtig gilt, auf 0.34 voreingestellt, der im Normalfall ausreichend ist.

3.1.4.5 Der Updatemechanismus

Thunderbird bietet für alle Plugins ein einfaches Verfahren, um diese auf dem neuesten Stand zu halten. Ein Update kann auf zwei verschiedene Weisen veröffentlicht werden. Eine Möglichkeit ist, dass das ursprüngliche Plugin über das Mozilla Portal [Moz] veröffentlicht wird, dann geschieht ein Update vollautomatisch. Doch bevor ein Plugin im

Mozilla Portal veröffentlicht wird, muss es im „Sandkasten“ einem „Review“ von Mozilla standhalten. Da dieser Weg sehr lange dauert, wird das Plugin meist auf fremden Webseiten veröffentlicht.
Wenn ein Plugin auf fremden Webseiten gehostet wird, muss in der Datei `install.rdf` eine Signatur hinterlegt werden. Mit Hilfe dieser Signatur wird gewährleistet, dass das Update auch tatsächlich vom richtigen Autor kommt. Ein neues Update muss nun immer vom Autor signiert werden. Die Datei `update.rdf` steuert dann das Updateverhalten von Thunderbird.
Updates werden automatisch von Thunderbird gesucht und gemeldet. Eine automatische Installation geschieht nicht.
Updates sind vollständige Plugins, deswegen löschen sie ihren Ordner in den Profilen und installieren sich selbst neu. Dies bringt einige Nachteile für den AttachmentChecker mit sich. Alle Datenbankdateien werden gelöscht und das bisher Gelernte ist nicht mehr verfügbar.

3.1.4.6 Internationalisierung

Das Plugin nutzt die Möglichkeit der Internationalisierung mit den von Thunderbird zu Verfügung gestellten Mitteln. Im Verzeichnis `local` wird für jede Sprache ein Verzeichnis mit dem jeweiligen Kürzel angelegt und dort Sprachdateien abgelegt. In JavaScript-Dateien werden hierfür Properties-Dateien gebraucht, die Zuweisungen enthalten.
In Abbildung 3.19 ist die Datei `attcheck_lang.properties` für Deutsch dargestellt, die die Variablen `yes`, `no` und `message` auf die deutschen Bedeutungen setzt.

```
1 ...
2 attcheck.yes=Ja
3 attcheck.no=Nein
4 attcheck.message=Diese Nachricht ist anhangverdächtig. Wollen
    Sie einen Anhang anfügen ?
5 ...
```

Abbildung 3.19: Ausschnitt attcheck_lang.properties (de)

Diese Datei gibt es im Ordner `en` ebenfalls und dort sind die Variablen mit der englischen Bedeutung belegt.
In XUL-Dateien ist die sprachabhängige Anzeige von Wörtern über DTD-Definitionen möglich. Am Beispiel des Kontextmenüs wird `attcheck_learn` mit der Bezeichnung des Menüeintrags in der Datei `attcheck_context.dtd` gesetzt (vgl. Abbildung 3.20).

```
1 <!ENTITY attcheck\_learn "Lerne␣AttachmentChecker">
```

Abbildung 3.20: Datei attcheck_context.dtd (de)

Wie auch bei den JavaScript-Übersetzungen müssen hier die englischen Übersetzungen im Verzeichnis en abgelegt sein.

3.2 Server

Der Server ist komplett in Python geschrieben. Da Python eine sehr große Standardbibliothek hat, können einige Funktionen direkt daraus entnommen werden. Bei der Implementierung wurde besonderen Wert darauf gelegt, dass es einfach ist, neue Algorithmen einzupflegen und zu testen.
Die in 3.21 abgebildete Architektur liegt dem Server zu Grunde.

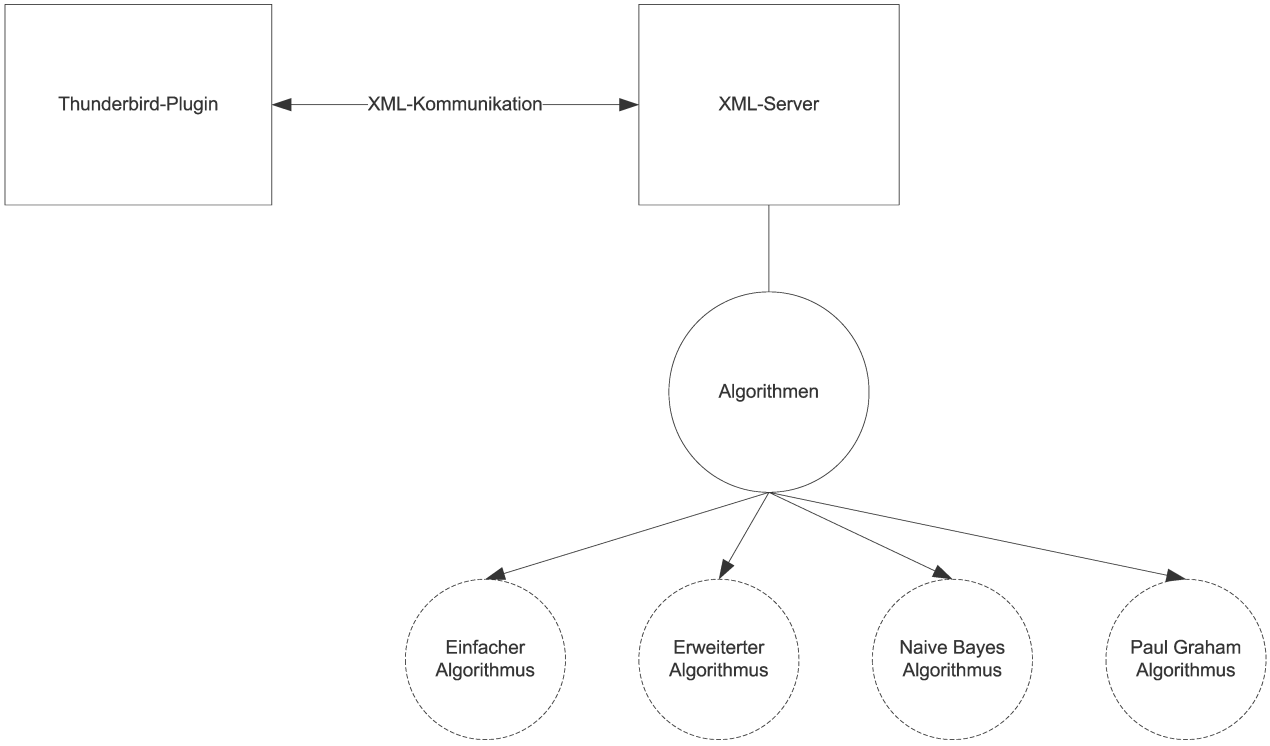

Abbildung 3.21: Architektur Server

Der XML-Server verarbeitet die Anfragen von Thunderbird und gibt die nötigen Daten an einen Algorithmus weiter. Der Algorithmus verarbeitet die Daten und sendet sie an Thunderbird zurück.

3.2.1 XML-Server

Der XML-Server nimmt alle Anfragen des Plugins entgegen und verarbeitet sie. Grundsätzlich werden alle Anfragen des Plugins per POST-Methode gesendet.
Die Unterscheidung, welche Art von Anfrage gestellt wurde, entscheidet die aufgerufene URL. Der Server kann unterscheiden zwischen

- Klassifikationsanfragen (`/check`),
- Lernen (`/learn`),
- nach dem Senden lernen (`/send`),
- Beenden (`/shutdown`) und
- Checkrun (`/checkrun`).

In Klammern sind die URL-Endung der Befehle angegeben.

3.2.1.1 Klassifikationsanfrage

Eine Klassifikationsanfrage wird gestellt, um zu erfahren, ob eine E-Mail anhangverdächtig ist. Sie besteht nur aus dem E-Mail-Text, der an einen Algorithmus zur Klassifikation übergeben wird.
Der Algorithmus klassifiziert und gibt je nach Ergebnis dann „True“ oder „False“ zurück. Dies entspricht der Antwort auf die Frage, ob die E-Mail anhangverdächtig ist. Diese Antwort wird dann an das Plugin zurückgemeldet.

3.2.1.2 Lernen

Das Lernen verläuft ähnlich einer Klassifikationsanfrage. Zusätzlich zum E-Mail-Text wird auch die Information, ob ein Attachment angehängt war, vom Plugin übertragen. Diese Klassifikationsinformation wird in der URL übergeben, die vom Plugin aufgerufen wird. Ist am Ende der URL ein „yes“ angehängt, bedeutet dies, dass ein Attachment an der E-Mail angehängt war und „no“, dass kein Attachment angehängt war.
Die Daten werden an einen der Algorithmen übergeben. Dieser Algorithmus führt die weitere Verarbeitung durch. An das Plugin wird als Antwort „OK“ gesendet.

3.2.1.3 Nach dem Senden lernen

Diese Funktion unterscheidet sich nicht von Lernen. Sie wurde lediglich getrennt implementiert, um eine spätere Veränderung der Funktion zu vereinfachen.

3.2.1.4 Beenden

Empfängt der Server den Beenden-Befehl, wird er sofort beendet. Eventuell offene Dateien werden gesichert.

3.2.1.5 Checkrun

Diese Funktion wird benötigt um zu überprüfen, ob der Server noch läuft. Läuft der Server antwortet er mit „OK“, ist er beendet wird ein Timeout gemeldet.

3.2.2 Algorithmen

In diesem Abschnitt werden Besonderheiten, die durch die Implementierung der Algorithmen aus Kapitel 2 entstanden sind, aufgeführt.

3.2.2.1 Wortextraktion

Damit die Algorithmen mit den E-Mails umgehen können, müssen sie in Wörter geteilt werden. Python bietet dafür den Befehl `findall`, der als Argument einen E-Mail-Text erwartet. `findall` wird auf einem regulären Ausdruck angewendet und gibt eine Liste von Zeichenketten zurück. Abbildung 3.22 veranschaulicht die Verwendung. [fin]

```
1 ...
2 tokens = self.config.TOKEN_RE.findall(data)
3 ...
```

Abbildung 3.22: Ausschnitt combined.py

3.2.2.2 Basis-Algorithmus

Da Python eine objektorientierte Sprache ist, ist die Vererbung ein Element das Zeit und Arbeit sparen kann.
Der Basis-Algorithmus stellt eine Art Interface für alle folgenden Algorithmen da. Eine Vorlage für die Funktionen zum

- Lernen,
- Klassifizieren,
- Abfragen der Datenbank und
- Erstellen des Caches

existieren.
Weil nicht jeder Algorithmus alle Funktionen benötigt, sind Standardimplementierung schon im Basis-Algorithmus enthalten. Der Algorithmus kann nicht alleine verwendet werden und hat deswegen keine Entsprechung in Kapitel 2.

3.2.2.3 Einfacher Algorithmus

Der einfache Algorithmus verwendet gesehene E-Mails nicht zur besseren Klassifizierung, braucht deswegen auch keine Datenbanken und erstellt auch kein Cache. Diese Funktionen werden vom Basis-Algorithmus übernommen.
Überschrieben wird die Funktion zum Klassifizieren. Hier ist festgelegt, welche Worte

zur Klassifizierung notwendig sind. Wird eines der Worte in der E-Mail gefunden, gibt der Algorithmus den Wert 0 für kein gefundenes Wort und 1 für gefundenes Wort zurück.
Umgewandelt wird 0 und 1 noch entsprechend in die Serverantwort „False“ oder „True“.

3.2.2.4 Erweiterter Algorithmus

Der erweiterte Algorithmus benötigt eine Wortliste und somit eine Datenbank. Ein Cache wird nicht genutzt. Die Datenbank besteht aus zwei einzelnen Dictionary-Datentypen, diese sind mit Hilfe von Hashes in Python implementiert. Dort werden die Vorkommnisse der Worte in Attachment-E-Mails und Noattachment-E-Mails festgehalten.
Das Lernen befüllt diese beiden Dictionaries. Um Laufzeit zu sparen, wenn mehrere E-Mails hintereinander gelernt werden, gibt es eine Funktion `build`, die nur aufgerufen wird, wenn eine Klassifizierung bevorsteht. Diese Funktion überprüft, ob durch das Lernen ein Wort in einer Menge häufiger vorkommt. Wird dies festgestellt, wird das jeweilige Wort aus der Wortliste hinzugefügt oder entfernt.
Die Abbildung 3.23 zeigt die `build`-Funktion.

```
1 def build(self):
2         for item in self.att.keys():
3             if (not self.noatt.has_key(item) and self.att[item
                ]>100) or (self.noatt.has_key(item) and self.att[
                item]>=(self.FACTOR*self.noatt[item] )):
4                 if item not in self.config.STARTSET:
5                     if item not in self.config.BLACKLIST :
6                         self.config.STARTSET = self.config.
                            STARTSET + [item]
7         for item in self.config.STARTSET:
8             if self.noatt.has_key(item) and self.att.has_key(
                item) and self.noatt[item]>=self.att[item]:
9                 self.config.STARTSET.remove(item)
```

Abbildung 3.23: Ausschnitt staticnumber.py

`FACTOR` entspricht dem Faktor v aus dem Algorithmus. Die Wortliste heißt `STARTSET`.
Leicht zu erkennen ist, dass es eine zusätzliche Variable `BLACKLIST` gibt. Dort kann angeben werden, welche Worte auf keinen Fall in die Wortliste eingefügt werden dürfen. Die Blacklist fand allerdings keine Anwendung und wurde nur zu Testzwecken benötigt.
Der Klassifizierungsprozess ist wie beim einfachen Algorithmus umgesetzt.

3.2.2.5 Naive Bayes Algorithmus

Der Naive Bayes Algorithmus benötigt alle Funktionen aus dem Basis-Algorithmus. Das Lernen besteht aus dem Befüllen der Dictonaries, wie beim erweiterten Algorithmus. Der Cache enthält zwei Dictionaries, die die Auftrittswahrscheinlichkeiten der Wörter in beiden Klassen erfassen. Der Cache wird aus Laufzeitgründen nur dann erzeugt, wenn eine Klassifizierung durchgeführt werden soll. Der Naive Bayes Algorithmus, wie in Kapitel 2 beschrieben, multipliziert alle Wahrscheinlichkeiten der Wörter in der zu klassifizierenden E-Mail miteinander. Da die Genauigkeit der Zahlen im Rechner begrenzt ist, kann es zu einem Unterlauf kommen. Dieser Unterlauf ist auf das Unterschreiten der kleinsten darstellbaren Zahl zurückzuführen, welche durch Multiplikation kleiner Wahrscheinlichkeiten durchaus auftreten kann. Dieses Problem kann verhindert werden, indem der Logarithmus angewendet wird. Zu beachten ist, dass nach dem Logarithmus-Gesetz, eine Multiplikation zur Addition wird (vgl. Abbildung 3.24). [CDM08]

```
1  def _check(self,c,test):
2          if self.att.tokencount==0 or self.noatt.tokencount==0:
3              p=math.log(0.5)
4          else:
5              p=math.log(float(c.tokencount)/(float(self.att.
                   tokencount)+float(self.noatt.tokencount)))
6          for item in test:
7              if c.has_key(item):
8                  p+=math.log(c[item])
9              else:
10                 p+=math.log(1/(float(self.att.tokencount)+float
                       (self.noatt.tokencount)))
11         return p
```

Abbildung 3.24: Ausschnitt bayes.py

Die Abbildung 3.24 zeigt auch, dass für ein Wort, das nicht existiert eine Wahrscheinlichkeit von $\frac{1}{\text{Anzahl der Wort}}$ angenommen wird. Außerdem wird für eine leere Klasse eine Wahrscheinlichkeit von 0.5 angenommen.
Die addierten Logarithmus-Werte müssen nicht rücktransformiert werden, da nur entschieden werden soll, welche Klasse den höheren Wert hat. Durch Normierung ist es möglich, den berechneten Wert der beiden Klassen wieder in einen Bereich zwischen 0 und 1 zu bringen. Dieser Wert entspricht nicht den geschätzten Wahrscheinlichkeiten. Es kann gezeigt werden, dass die Rechnung numerisch stabil ist.

3.2.2.6 Paul Grahams Algorithmus

Paul Grahams Algorithmus war ursprünglich in Lisp geschrieben und musste in Python übersetzt werden.[Gra02]
Der Algorithmus besitzt eine Datenbank, die die vorberechneten Werte der Worte enthält. Das Lernen erfolgt analog dem Naive Bayes Algorithmus und unterscheidet sich nur dadurch, dass die Anzahl der gelernten E-Mails zusätzlich gezählt werden.
Laufzeitoptimierungen wurden beim Lernprozess durchgeführt, da die Berechnung der Datenbank nur dann notwendig ist, wenn eine Klassifizierung durchgeführt werden soll. Der Wiedergabewert der Klassifizierung liegt zwischen 0 und 1.

3.2.2.7 Combined-Algorithmus

Der Combined-Algorithmus stellt eine Kombination des Paul Graham und Naive Bayes Algorithmus da. Da beide Algorithmen auf einer Menge von Worten arbeiten, kann der Algorithmus so implementiert werden, dass Speicherplatz eingespart wird, indem beide die gleiche Lernmenge nutzen.
Zuerst wird der Paul Graham Algorithmus und danach Naive Bayes Algorithmus ausgeführt. Eine andere Reihenfolge der Ausführung wirkt sich nicht auf das Klassifikationsergebnis aus.

3.3 Alternative Implementierungen

Zu Beginn der Implementierungsphase der Arbeit musste ich mich für einen von drei Architekturentwürfen entscheiden. Zur Auswahl stand eine Umsetzung

- nur mit JavaScript und XUL,
- mit PyXPCOM,
- und als Server-Client.

Die Server-Client Architektur wurde oben bereits ausführlich erklärt und ist umgesetzt.

Eine Implementierung nur mit JavaScript und XUL ist eine gute Alternative, wenn keine weiteren Analysen durchgeführt werden müssen. Eine Analyse ist schwer ohne einen Interpreter möglich, da hier immer Thunderbird als ausführende Instanz hätte laufen müssen. Änderungen an Plugins werden nach einem Neustart gültig und die Fehleranalyse ist nicht so genau, wie bei einem Interpreter oder Compiler. Da in dieser Arbeit zu Anfang nicht klar war, welcher Algorithmus implementiert wird, mussten sehr viele Tests durchgeführt werden. Diese hätten in der Kombination mehr Zeit in Anspruch genommen, deswegen fiel diese Alternative weg.

PyXPCOM ist die XPCOM-Anbindung an Python. Der Vorteil einer solchen Anbindung

ist, dass das Programm in Python geschrieben wird und es dadurch testbar geworden wäre, weil es auch ohne Thunderbird ausführbar bleibt. Allerdings ist PyXPCOM so schlecht dokumentiert, dass diese Umsetzung nicht mit vertretbarem Zeitaufwand zu schaffen gewesen wäre.
Es gibt ein Projekt, welches versucht, die Einbindung zu vereinfachen. Dieses Projekt heißt pyxpcomext und ist als Erweiterung für Thunderbird verfügbar. Die dort verfügbaren Testprogramme waren allerdings ebenfalls sehr schlecht durchschaubar. Aus diesem Grund wurde von einer Implementierung des AttachmentCheckers auf diese Weise abgesehen. [pyx]

Kapitel 4

Evaluierung

Die Evaluierung zeigt, wie gut die Algorithmen aus Kapitel 2 klassifizieren und damit das Attachment-Problem lösen.
Da die Algorithmen nicht ohne weiteres vergleichbar sind, benötigt man Evaluierungsmaße, die bei jedem Algorithmus bestimmbar sind. In diesem Kapitel werden Evaluierungsmaße aus dem Information Retrieval und aus der Signalerkennungstheorie verwendet. Diese Maße lassen sich nur durch geeignete Testdatensätze bestimmen.

4.1 Evaluierungsmaße

Aus dem Information Retrieval sind die Maße Accuracy, Precision, Recall und Fallout bekannt. Diese lassen sich aus den Werten der Konfusionsmatrix berechnen
(vgl. Abbildung 4.1). [con08]
In der Konfusionmatrix sind die vier Werte

- True Positive - E-Mail mit Attachment, die richtig klassifiziert wurde,
- False Positive - E-Mail ohne Attachment, die falsch klassifiziert wurde,
- True Negative - E-Mail ohne Attachment, die richtig klassifiziert wurde und
- False Negative - E-Mail mit Attachment, die falsch klassifiziert wurde

enthalten.

	Attachment erkannt	kein Attachment erkannt
hat Attachment	True Positive	False Negative
kein Attachment	False Positive	True Negative

Abbildung 4.1: Konfusionsmatrix

Die ROC-Kurve sowie der damit verbundene AUC-Wert kommen aus der Signalerkennungstheorie und benötigen lediglich die Werte, die der Klassifizierer ausgibt.

4.1.1 Accuracy

Accuracy heißt übersetzt Genauigkeit. Da die Übersetzung für Precision ebenfalls Genauigkeit ist, werden die englischen Begriffe der Maße verwendet. In der Fachliteratur findet man oft den Ausdruck Korrektklassifikationsrate als Übersetzung für Accuracy. Die Accuracy beschreibt das Verhältnis zwischen korrekt erkannten E-Mails zu allen E-Mails. Damit ergibt sich

$$\text{Accuracy} = \frac{\text{True Positive} + \text{True Negative}}{\text{True Positive} + \text{True Negative} + \text{False Negative} + \text{False Positive}}.$$

Dieses Maß, sollte nicht alleine betrachtet werden, weil bei einem Zweiklassenproblem die Aufteilung der Klassen wichtig ist. Sind die beiden Klassen ungleich verteilt, muss man dies in der Bewertung und Interpretation des Ergebnisses beachten.

Fallbeispiel:
Nehmen wir an, dass ein Klassifizierer keine Attachments erkennt und der Datensatz 100 E-Mails hat, wovon zehn ein Attachment haben. Daraus ergibt sich

$$\text{Accuracy} = \frac{0 + 90}{0 + 90 + 10 + 0} = 0.9.$$

Die Accuracy bescheinigt dem Klassifizierer, dass er zu 90% richtig klassifiziert hat. Betrachtet man sich allerdings den Anteil der Attachments, stellt man fest, dass der Klassifizierer genau um den Prozentsatz der Attachments falsch klassifiziert.
Dieses Beispiel verdeutlicht, dass das alleinige Betrachten der Accuracy nicht ausreichend ist.

4.1.2 Precision

Precision heißt übersetzt auch Genauigkeit und gibt an, wieviele E-Mails mit Attachment von allen gemeldeten Attachments richtig erkannt wurden.

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}}$$

Ein ähnliches Problem wie bei Accuracy zeichnet sich ab, wenn dieses Maß alleine betrachtet wird. Erkennt der Klassifizierer nur ein Attachment und meldet dann nie wieder ein Attachment ist der Precision-Wert sehr hoch.

Fallbeispiel:
Nehmen wir an, dass ein Klassifizierer genau ein Attachment erkennt und die Menge 100 E-Mails hat, wovon zehn ein Attachment haben. Daraus ergibt sich

$$\text{Precision} = \frac{1}{1+0} = 1.$$

Der Precision-Wert bescheinigt dem Klassifizierer, dass er 100 % der Attachment-E-Mails richtig klassifiziert hat. Betrachtet wurde jedoch nicht, wie viele erkannte Attachments im Vergleich zu den gesamten Attachments enthalten sind, und damit ist der Wert alleine auch nicht aussagekräftig.

4.1.3 Recall

Ein weiteres Maß ist der Recall, der das Verhältnis zwischen der Anzahl der korrekt gefundenen Attachments zu allen Attachments angibt. Es ist definiert durch

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}}.$$

Der Recall ist sehr hoch, wenn man annimmt, dass alle E-Mails ein Attachment haben.

Fallbeispiel:
Nehmen wir an, dass ein Klassifizierer alle E-Mails als anhangverdächtig erkennt und die Menge 100 E-Mails hat, wovon zehn ein Attachment haben. Daraus ergibt sich

$$\text{Recall} = \frac{10}{10+0} = 1$$

Der Recall bescheinigt dem Klassifizierer, dass er 100 % der Attachment-E-Mails richtig klassifiziert hat. Betrachtet wurde nicht, wieviele E-Mails ohne Attachment falsch klassifiziert wurden.

4.1.4 Fallout

Fallout ist das Verhältnis zwischen fälschlicherweise als anhangverdächtig eingestuften E-Mails zu allen E-Mails ohne Attachment:

$$\text{Fallout} = \frac{\text{False Positive}}{\text{False Positive} + \text{True Negative}}.$$

Werden keine Attachments erkannt, ist das Maß sehr niedrig. Im Gegensatz zu allen anderen Maßen ist es gut, wenn Fallout niedrig ist.

Fallbeispiel:
Nehmen wir an, dass ein Klassifizierer keine Attachments erkennt und die Menge 100 E-Mails hat, wovon zehn ein Attachment haben. Daraus ergibt sich

$$\text{Fallout} = \frac{0}{0 + 90} = 0.$$

Dieses Maß wird in der Arbeit nur in den Ergebnistabellen im Anhang angegeben.

4.1.5 Precision-Recall-Diagramm

Die beiden Maße Precision und Recall können nicht alleine betrachtet werden, daher stellt das Precision-Recall-Diagramm eine Kombination beider Maße dar.
Auf der x-Achse zeichnet man den Recall und auf der y-Achse den Precision-Wert ein. Es werden alle Precision- und Recall-Werte ohne Beachtung des Schwellwertes aufgetragen. Das bedeutet, dass man einen Algorithmus mit verschiedenen Schwellwerten, ab dem die E-Mail als anhangverdächtig gilt, ausführen muss. Man erhält beispielsweise Werte, wie in Tabelle 4.1 abgebildet.

Schwellwert	Precision	Recall
0.0	0.1	1.0
0.2	0.3	0.9
0.4	0.7	0.8
0.6	0.8	0.7
0.8	0.9	0.1
1.0	1.0	0.0

Tabelle 4.1: Beispiel PR-Diagramm

Trägt man diese Punkte in ein Koordinatensystem ein und verbindet sie, erhält man das Precision-Recall-Diagramm (vgl. Abbildung 4.2). Der höchste Wert im Diagramm an dem der Precision-Wert gleich dem Recall-Wert ist, wird Precision-Recall-Breakeven-Punkt genannt. In dem Beispiel wäre das der Punkt mit einem Precision- und Recall-Wert von 0.75. Umso besser der zugrunde liegende Algorithmus, desto besser ist dieser Wert. Optimal ist ein Precision-Recall-Breakeven-Punkt von 1.0, der aussagt, dass der Algorithmus bei diesem Schwellwert alle E-Mails korrekt klassifiziert.

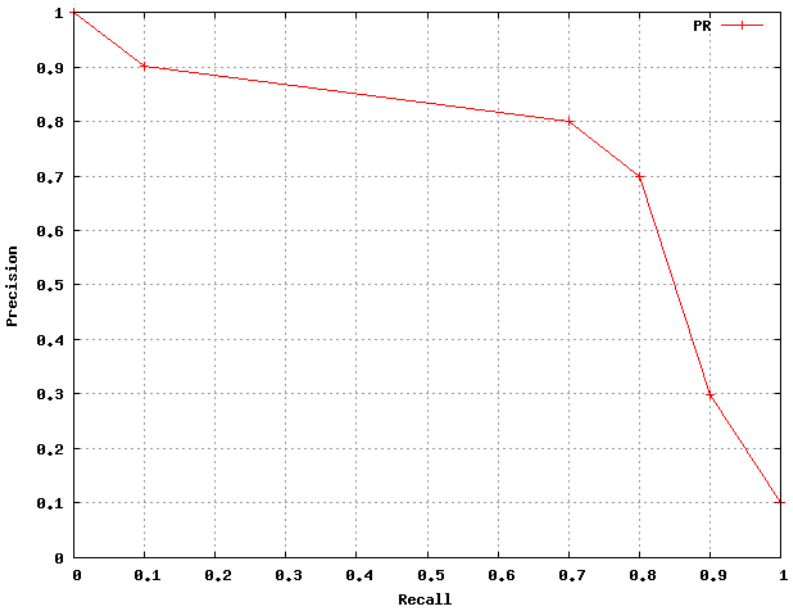

Abbildung 4.2: Precision-Recall-Diagramm

4.1.6 ROC-Kurve

ROC steht für Receiver Operating Characteristic und kommt aus der Signalerkennungstheorie. Ursprünglich wurde die ROC-Kurve benutzt um Radarbilder im zweiten Weltkrieg zu analysieren.
Mit Hilfe der ROC-Kurve kann im maschinellen Lernen die Trennung eines Zweiklassenproblems betrachtet werden. Der Schwellwert wird hierbei nicht beachtet.
Auf der x-Achse trägt man die sogenannte False Positive Rate auf und auf der y-Achse die True Positive Rate. Die False Positive Rate ist definiert durch

$$FPR = \frac{n}{N},$$

wobei N die Anzahl der E-Mails ohne Attachment im Datensatz und n die Anzahl der abgedeckten E-Mails ohne Attachment ist. Analog dazu ist die True Positive Rate definiert als

$$TPR = \frac{p}{P},$$

wobei P die Anzahl der E-Mails mit Attachment im Datensatz und p die Anzahl der abgedeckten E-Mails mit Attachment darstellt.
Die Schrittweite einer E-Mail auf der x-Achse ist $\frac{1}{N}$ und ein Schritt auf der y-Achse $\frac{1}{P}$. Gezeichnet wird eine ROC-Kurve, indem man die E-Mails nach ihrem Klassifikationswert, den ein Algorithmus wiedergegeben hat, sortiert und diese Werte dann vom größten Klassifikationswert bis zum kleinsten Wert einzeichnet. Diese Werte zeichnet man so ein, dass eine E-Mail mit Attachment einen Schritt auf der y-Achse bedeutet und eine

E-Mail ohne Attachment ein Schritt auf der x-Achse. Abbildung 4.3 zeigt die Sortierung beispielhaft mit den E-Mails A, B, C, D, E, F.

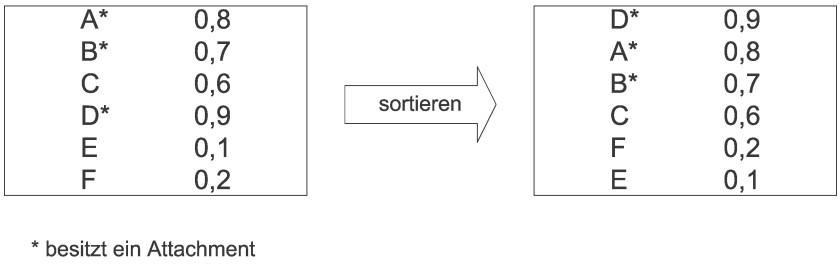

Abbildung 4.3: Sortierung für ROC-Kurve

Die E-Mails mit den Sternen — also A, B, D — haben Attachments und die anderen verbleibenden haben keine. Die Sortierung ist abgeschlossen, wenn die Klassifikationswerte absteigend sortiert sind.
Daran ist zu erkennen, dass eine optimale Kurve zuerst alle E-Mails mit Attachments in der Liste aufführt und danach alle E-Mails ohne Attachments. Zeichnerisch bedeutet dies, dass zuerst alle Schritte auf der y-Achse gemacht werden und danach alle auf der x-Achse. Dann spricht man auch von einer perfekten Trennung.
Die Kurve der Abbildung 4.5 ist gut, weil sie erst bei 0.7 den ersten Schritt in x-Richtung macht.

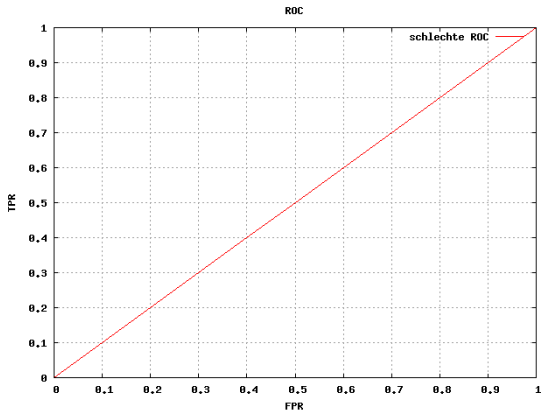

Abbildung 4.4: Schlechte ROC-Kurve

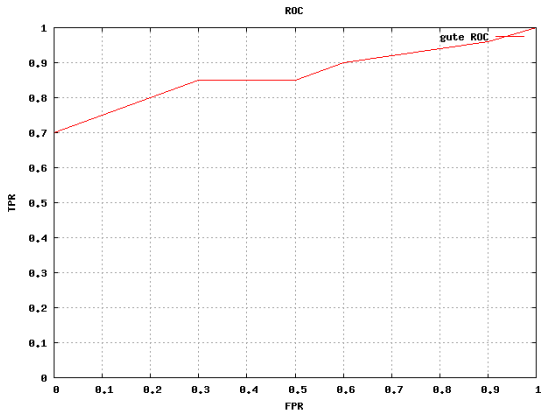

Abbildung 4.5: Gute ROC-Kurve

Die Abbildung 4.4 zeigt eine sehr schlechte Kurve. Eine solche Kurve bedeutet, dass die E-Mails willkürlich klassifiziert wurden. Durch die Sortierung der Liste wird keine Trennung zwischen E-Mails mit Attachment und E-Mails ohne Attachment sichtbar.
Abhängig vom verwendeten Testverfahren ist es manchmal notwendig, mehrere ROC-Kurven zu einer Durchschnittskurve zusammenzufassen. Dazu muss man zuerst für jede Kurve diskrete Werte an den Stellen 0.0, 0.1, 0.2,...,0.9 berechnen. Die Berechnung dieser Werte kann durchgeführt werden, indem abschnittsweise eine Gerade berechnet

und der Wert dazwischen interpoliert wird. Die Werte zwischen einzelnen Punkten sind im Regelfall nicht definiert.

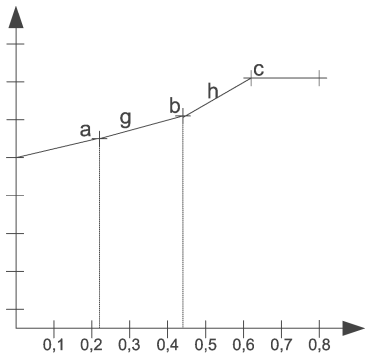

Abbildung 4.6: Beispiel ROC Geradenberechnung

Um in Abbildung 4.6 die Punkte 0.3 und 0.4 zwischen den Punkten a und b zu bestimmen, benötigt man die Gerade g. Für Punkt 0.5 und 0.6 jedoch die Gerade h. Ein Abschnitt ist also mit zwei Punkten begrenzt.
Die Berechnung einer Geraden zwischen zwei Punkten kann mit der Gleichung

$$f(x) = m\dot{x} + b$$

bestimmt werden. Um die Steigung m zu ermitteln berechnet man

$$m = \frac{y_2 - y_1}{x_2 - x_1},$$

wobei y_1 und x_1 den Punkt links und y_2 und x_2 den Punkt rechts darstellen. Der y-Achsenabschnitt b lässt sich direkt ermitteln durch

$$b = \frac{y_2}{m \cdot x_2}.$$

Sind von allen Kurven die diskreten Werte ermittelt, bildet man das arithmetische Mittel über jede Stelle und kann die erhaltenen Punkte in ein neues Koordinatensystem einzeichnen. Die eingezeichnete Kurve entspricht der Durchschnittskurve.
Eine alternative Möglichkeit ist, dass man den letzten y-Wert auf die x-Achse überträgt. Die Abbildung 4.7 verdeutlicht, dass der Punkt 0.3 und 0.4 den y-Wert des Punktes a annehmen würde. Wie zu erkennen, hat die resultierende Kurve den Nachteil, dass die Kurve bei wenigen Punkten eine steile Steigung aufweist. Aus diesem Grund wurde bei allen Durchschnittsberechnungen die erste Variante gewählt.

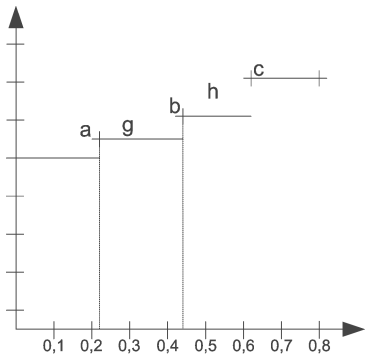

Abbildung 4.7: Beispiel ROC alternative Berechnung

Die Fläche unter der ROC-Kurve wird AUC (=Area under curve) genannt. Der AUC-Wert bei einer perfekten ROC-Kurve entspricht 1.0, der schlechteste Wert dementsprechend 0.5. Zwischen 0.5 und 1.0 gibt es noch weitere Abstufungen, die wie folgt eingeteilt werden können [auc]:

- 1.0-0.9 Sehr gute Trennung
- 0.9-0.8 Gute Trennung
- 0.8-0.7 Befriedigende Trennung
- 0.7-0.6 Schlechte Trennung
- 0.6-0.5 Verfahren schlägt fehl

Um den AUC zu berechnen gibt es zwei Möglichkeiten. Für die erste Möglichkeit benötigt man die zur Erstellung der ROC-Kurve benutzte und sortierte Liste. Der Rang r_i einer E-Mail ist die Nummer des Listeneintrags (vgl. Abbildung 4.8).

Rang		
1	D*	0,9
2	A*	0,8
3	B*	0,7
4	C	0,6
5	F	0,2
6	E	0,1

* besitzt ein Attachment

Abbildung 4.8: Beispiel ROC Rang

Man bestimmt die Summe aller Ränge S_- der E-Mails ohne Attachment. Außerdem die Anzahl aller im Testdatensatz vorkommenden E-Mails mit Attachments P und ohne Attachments N. Beim Berechnen von

$$S_- = \sum_i r_i$$

ist es wichtig, dass die E-Mail mit dem höchsten Klassifikationswert den Rang $r = 1$ hat. Ist S_- berechnet, kann

$$AUC = \frac{S_- - N \cdot (N+1)/2}{P \cdot N}$$

berechnet werden, wobei angenommen wird, dass die Klassifikationswerte eindeutig sind. [DJH01]
Beim einfachen und erweiterten Algorithmus sind die Klassifikationswerte nicht abgestuft. Die beiden Algorithmen geben nur eine 0 oder eine 1 zurück, wodurch die Sortierung unter diesen Werten willkürlich ist. Beim Zeichnen der Kurve werden diese Punkte zusammengefasst, was allerdings bei der Bestimmung des AUC Probleme machte. Die Ränge der E-Mails ohne Attachments können so beliebig in den beiden Listenteilen — oberer Listenteil mit nur Einsen, unterer Listenteil mit nur Nullen — umsortiert werden.
Um in diesem Fall den AUC trotzdem zu berechnen, wird die Fläche unter der Kurve stückweise berechnet. In Abbildung 4.9 ist diese Methode veranschaulicht. Dort erkennt man, dass die Abschnitte eins bis fünf addiert werden müssen, damit die Gesamtfläche berechnet ist.

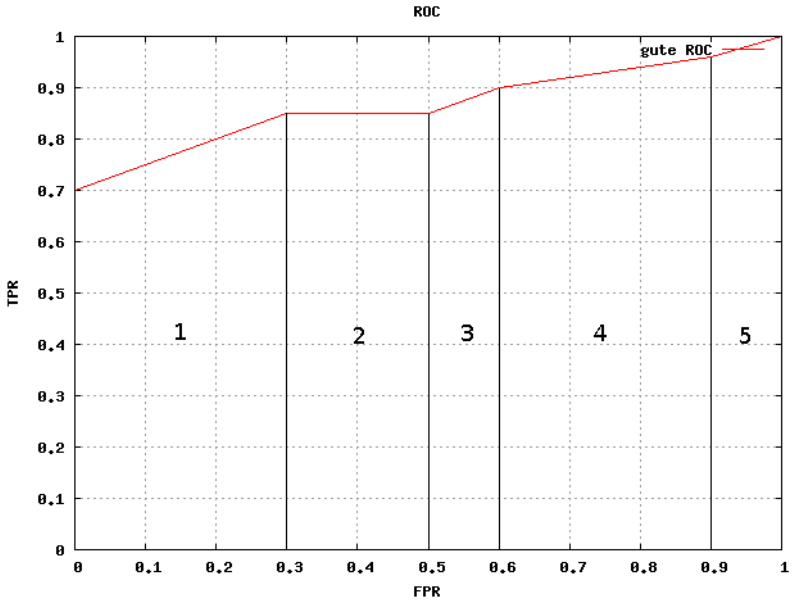

Abbildung 4.9: Berechnung AUC

Da die Geradengleichungen der einzelnen Flächen durch die Berechnung der Durchschnittskurve bereits bestimmt ist, muss diese integriert werden. Das Integral ist

$$F(x) = \int_a^b m \cdot x + b = \left[\frac{1}{2} \cdot m \cdot x^2 + b \cdot x\right]_a^b.$$

Daraus folgt

$$AUC = \sum F(b) - F(a),$$

wobei a und b die x-Werte der Punkte im Diagramm sind.
Die Durchschnittsfläche mehrerer ROC-Kurven berechnet man durch das arithmetische Mittel der AUC aller Kurven.

4.2 Verfahrensweise

Da die Evaluierung der Datensätze nicht ganz einfach ist, wird in diesem Kapitel erklärt, wie getestet wurde. Benutzt wurde eine Cross-Validation und einige Testprogramme.

4.2.1 Cross-Validation

Die Cross-Validation wird angewendet, wenn kein expliziter Trainings- und Testdatensatz gegeben ist. Es gibt also keinen ausgezeichneten Datensatz von E-Mails, der nur zum Testen und ein anderer, der nur zum Trainieren genutzt wird.
Der zur Verfügung stehende Datensatz wird in K Partitionen aufgeteilt. Von diesen K Partitionen wird eine Partition als Testdaten und die restlichen $K-1$ Partitionen als Trainingsdaten verwendet. Die Cross-Validation wird nun K-mal wiederholt, sodass jede der K Partitionen genau einmal als Testdaten benutzt wurde (vgl. Abbildung 4.10).

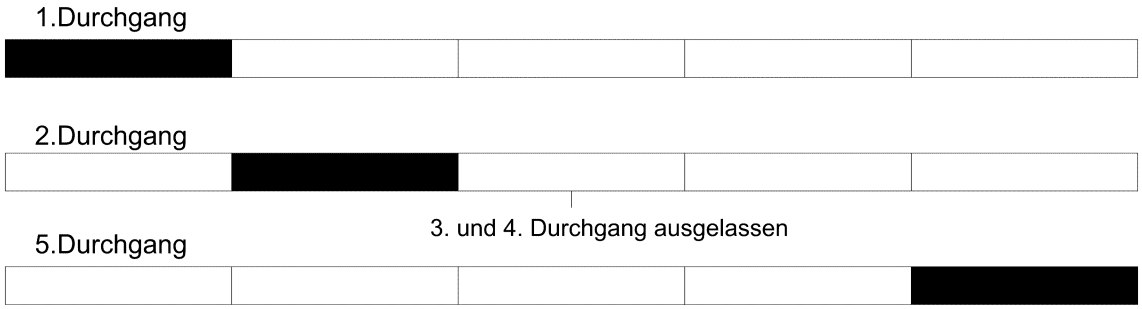

Abbildung 4.10: Cross-Validation mit $K = 5$

In der Abbildung 4.10 wurde vereinfacht $K = 5$ gewählt.
Da im Attachment-Problem davon auszugehen ist, dass es wesentlich mehr E-Mails ohne Attachments als mit Attachments gibt, muss sichergestellt werden, dass die Testdaten stratifiziert sind. Dies bedeutet, dass der Datensatz zufällig so zusammengesetzt wird, dass alle Testdaten gleichverteilt sind. Es darf kein Durchgang geben, indem die Testdaten kein Attachment haben, dadurch würden Mittelwerte verfälscht.
Sichergestellt wird die Stratifizierung der Testdaten mit einer Teilung der Attachment-Beispiele und den Noattachment-Beispielen vom Datensatz. Nun teilt man die erhaltenen Mengen in K zufällige Blöcke zu je n Elementen. Die Anzahl der Elemente in den Mengen ist $\frac{\text{Anzahl der E-Mails}}{K}$ und damit für jede Menge unterschiedlich . Weiter verbindet

man je einen Block Attachment-Beispiele mit einem Block Noattachment-Beispiele und randomisiert sie. Es darf kein Beispiel doppelt verwendet werden.

Fallbeispiel:
Nehmen wir an, die Menge hat 100 E-Mails, wovon zehn ein Attachment haben. Daraus ergibt sich, dass die Attachment-Menge 90 E-Mails und die Noattachment-Menge zehn E-Mails haben.
Die entstandenen Testdaten bestehen dann jeweils aus einer E-Mail mit Attachment und neun E-Mails ohne Attachment. Die Reihenfolge ist zufällig.

In den Tests wurde ausschließlich die 10-Fold Cross-Validation (10 Partitionen) genutzt.

4.2.2 Testprogramme

Um die Menge an E-Mails bearbeiten zu können, mussten einige Testprogramme erstellt werden. Manuelle Tests hätten zuviel Zeit in Anspruch genommen. Der Attachment-Checker musste an einer Stelle ebenfalls angepasst werden.

4.2.2.1 Anpassung AttachmentChecker

Der AttachmentChecker ist in der ursprünglichen Version nicht dafür ausgelegt E-Mails in eine Datei zu speichern. Um allerdings Analysen auf E-Mails auszuführen, wurde der AttachmentChecker so erweitert, dass er E-Mails als Klartext auf der Festplatte speichern kann. Diese Funktion ist über Thunderbird nicht aktivierbar. Dazu ist das Starten des Python-Servers mit der Befehlszeile
`python server.py 9999 0.34 . debug \verzeichnis\datei`
nötig. Dies wurde so gewählt, damit der Modus nicht aus Versehen aktiviert werden kann. Lernt man die gesendeten E-Mails mit Thunderbird, legt der AttachmentChecker im Verzeichnis `\verzeichnis` Dateien mit dem Präfix `datei` an. Damit keine Datei gelöscht wird, werden sie automatisch fortlaufend durchnummeriert. Die gespeicherten E-Mails haben dann die Form wie in Abbildung 4.11 gezeigt. „True“ bzw. „False“ gibt

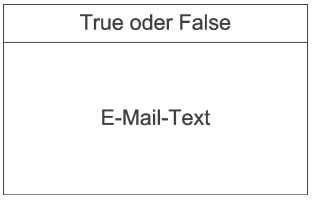

Abbildung 4.11: Aufbau E-Mail zur Analyse

an, ob ursprünglich ein Attachment angehängt war oder nicht.

4.2.2.2 Tester

Der Tester führt die Analyse der im letzten Abschnitt entstandenen Dateien durch. Gestartet wird er durch
`python tester.py a 10`
Die Befehlszeile bedeutet, dass der Tester eine 10-Fold-Cross-Validation durchführt und dabei alle Algorithmen testet. Mögliche Kürzel für den ersten Parameter sind:

- a — Alle Algorithmen
- b — Naive Bayes Algorithmus
- g — Paul Graham Algorithmus
- d — Erweiterter Algorithmus
- s — Einfacher Algorithmus
- c — Combined-Algorithmus

Es kann mit dem Tester eine beliebige K-Fold Validation durchgeführt werden. Beschränkend gilt, dass die Anzahl der Partitionen nicht kleiner sein darf als die Anzahl der E-Mails mit Attachment. Das heißt, dass jede Partition mindestens eine E-Mail mit Attachment hat.

Der Tester bearbeitet die E-Mails nach folgendem Schema

1. Teile die E-Mails in einen Teil mit und ohne Attachments
2. Erstelle stratifizierte Testmenge
3. Trainiere Algorithmus auf den verbleibenden E-Mails
4. Teste Testmenge

Dieser Ablauf wird solange ausgeführt, bis jede E-Mail einmal im Testset enthalten war. Bei einer 10-Fold-Cross-Validation ist dies zehn Mal. Außerdem muss die Cross-Validation für jeden Algorithmus durchgeführt werden.
Die Ergebnisse des Paul Graham, Naive Bayes und Combined-Algorithmus werden als ROC-Datei gespeichert. Eine ROC-Datei enthält von allen E-Mails die Klassifizierungswerte.
Beim einfachen Algorithmus gibt es nur ein Ergebnis, welches mit True Positive, False Positive, True Negative und False Negative in einer Datei gesichert wird.
Beim erweiterten Algorithmus werden immer alle Parameter von zwei bis 15 getestet, weil diese nicht aus einer ROC-Datei ablesbar sind.
Sind alle Algorithmen getestet, können Einzelwerte mit Hilfe des ROC-Converters extrahiert werden.

4.2.2.3 ROC-Converter

Der ROC-Converter konvertiert die vom Tester erstellten ROC-Dateien in einzelne Auswertungen. Diese Auswertungen sind die Basis zur Berechnung der Accuracy-, Recall-, Precision- und Fallout-Werte. Der ROC-Converter muss dafür auf jede ROC-Datei, die vom Tester erstellt wurde, einzeln angewendet werden. Der Befehl lautet:
`python rocconverter.py c0roc.log c0`,
damit wird die Datei `c0roc.log` in eine Datei `c0.ana` umgewandelt. Die entstandene Datei kann in eine Tabellenkalkulation importiert werden. Die kompletten Tabellen dieser Arbeit befinden sich im Anhang.

4.2.2.4 Analyser

Der Analyser erstellt aus den ROC-Dateien die Informationen, die von GNUPlot benötigt werden um die ROC-Kurven zu zeichnen. In diesem Programm werden Wertelisten für alle Kurven der Durchläufe erstellt, diese mit diskreten Werten interpoliert und schließlich eine Durchschnittskurve ermittelt. Für GNUPlot werden zum automatischen Zeichnen Skripts erstellt.
Außerdem wird der AUC-Wert mit beiden in diesem Kapitel erklärten Verfahren berechnet.

4.2.2.5 Wordtester

Der Wordtester erstellt zufällige Worte und fügt sie der gleichen Datenstruktur zu, die auch bei den Algorithmen verwendet wird. Dadurch ist es möglich den Speicherverbrauch der resultierenden Datenbanken zu messen. Der Wordtester ist in der Lage die Ergebnisse als LOG-Dateien zu exportieren, womit sich ein Diagramm zeichnen lässt.

4.2.2.6 Renamer

Der Renamer benennt E-Mail-Klartext-Dateien in die Benennung um, wie sie vom Tester benötigt werden. Dieses Helferprogramm wird benötigt, wenn das Lernen im Debugmodus abgebrochen wurde und neu gestartet wird. Die E-Mails werden dann einheitlich benannt und durchnummeriert. Gewöhnlich wird es in Kombination mit den oberen Programmen nicht benötigt.

4.2.2.7 Crawler

Der Crawler wurde zu Anfang der Arbeit verwendet um das Enron-Datenset (vgl. Abschnitt 4.3.1) nach Merkmalen, die auf ein Attachment hinweisen, zu untersuchen. Da das Enron-Datenset sehr viele E-Mails enthält, ist eine manuelle Durchsicht sehr langwierig.

4.2.2.8 GNUPlot

GNUPlot ist nicht im Rahmen dieser Arbeit entstanden. Es ist ein frei erhältliches Programm, dass auf der Webseite des Herstellers heruntergeladen werden kann. Mit diesem Programm lassen sich Diagramme und Plots jeder Art erstellen und speichern [GNU].

4.3 Ergebnisse

Die Ergebnisse stellen dar, welcher Algorithmus auf den Testdatensätzen und Kontrolldatensätzen am besten klassifiziert hat.
Da alle Datensätze in deutscher Sprache sind wurden für den einfachen Algorithmus die Worte: 'anhang', 'anhänge', 'datei', 'dateien', 'fotos', 'version', 'versionen', 'entwurf', 'entwürfe', 'anbei' gewählt. Die Wortmenge ist dabei willkürlich gewählt worden. Der erweiterte Algorithmus ist in den Tabellen immer mit Faktor 2 angegeben. Die Werte von Paul Graham sind bei einem Schwellwert von 0.51 und Naive Bayes mit 0.5 ermittelt worden. Im kombinierten Algorithmus ist Naive Bayes mit doppeltem Gewicht eingegangen, der Schwellwert beträgt 0.34.
Weitere Ergebnisse und Diagramme sind im Anhang zu finden.

4.3.1 Datensätze

Das Finden geeigneter Datensätze ist schwer, da nur wenige E-Mail-Nutzer freiwillig private E-Mails aus der Hand geben. Der bekannte Enron-E-Mail-Korpus [Coh05] kann leider nicht eingesetzt werden, da alle Informationen, die nötig wären, um herauszufinden, ob ein Attachment ursprünglich angehängt war, entfernt wurden. Durch die Veränderung der Headerdaten der E-Mails sind diese leider für eine Klassifizierung unbrauchbar.
Aus diesem Grund standen im Rahmen dieser Arbeit lediglich drei Datensätze — auch Testdatensätze genannt — zur Optimierung der Algorithmen zur Verfügung:

- 1. Datensatz mit 453 E-Mails in deutscher Sprache, davon hatten 74 E-Mails mindestens ein Attachment,
- 2. Datensatz mit 531 E-Mails in deutscher Sprache, davon hatten 113 E-Mails mindestens ein Attachment,
- 3. Datensatz mit 2245 E-Mails in deutscher Sprache, davon hatten 732 E-Mails mindestens ein Attachment.

Zu den Datensätzen waren mir jederzeit die Klartexte verfügbar um weitere Tests durchzuführen. Für Tests, die Veränderungen an Algorithmen evaluieren, sind die Klartexte nötig.
Zwei weitere Datensätze — auch Kontrolldatensätze genannt — standen mir zum Test der Ergebnisse zur Verfügung:

- 4. Datensatz mit 282 E-Mails in deutscher Sprache, davon hatten 70 E-Mails mindestens ein Attachment,
- 5. Datensatz mit 922 E-Mails in deutscher Sprache, davon hatten 96 E-Mails mindestens ein Attachment.

Auf diesen Datensätze wurden „blinde“ Tests durchgeführt. Das bedeutet, dass mir der Inhalt dieser E-Mails unbekannt ist und ich keine Optimierung der Algorithmen darauf durchführen konnte.
Alle Datensätze sind von Privatpersonen und können im Rahmen dieser Arbeit nicht veröffentlicht werden.

4.3.2 Accuracy

Die Accuracy-Werte geben die Genauigkeit auf den Datensätzen an. Ein Accuracy-Wert von 100% würde bedeuten, dass alle E-Mails korrekt klassifiziert sind. Die beiden Klassen im Attachment-Problem sind nicht gleichverteilt, deswegen ist in den Tabellen die „Nie Attachment“-Regel angegeben, die angibt, wie hoch der Accuracy-Wert ist, wenn niemals ein Attachment gemeldet wird.
Der einfache und der erweiterte Algorithmus schneiden bei den Accuracy-Werten am schlechtesten ab. Diese Algorithmen sind nur leicht besser oder leicht schlechter als die „Nie Attachment“-Regel. Daraus lässt sich schließen, dass beide Algorithmen keinen Vorteil für den Benutzer bringen.

Accuracy	Datensatz 1	Datensatz 2	Datensatz 3
Einfacher Algorithmus	76.38%	79.66%	71.00%
Erweiterter Algorithmus (2)	78.81%	80.04%	69.53%
Paul Graham (0.51)	87.64%	90.21%	85.84%
Naive Bayes (0.5)	87.64%	89.64%	88.95%
Combined (2:1,0.34)	93.82%	90.77%	88.55%
Nie Attachment	83.66%	78.68%	67.32%

Tabelle 4.2: Accuracy - Testdatemsätze

Der Algorithmus von Paul Graham und der Naive Bayes-Algorithmus liegen bei den Accuracy-Werten nahe beieinander und sind immer oberhalb der „Nie Attachment“-Regel. Der beste Algorithmus ist hier der Combined-Algorithmus. Allerdings ist auffällig, dass er im Datensatz 3 schlechter als der Naive Bayes abschneidet. Dieser Ausreißer könnte durch das „schlechte“ Ergebnis des Paul Grahams verursacht worden sein. Die Accuracy-Werte zeigen, dass im besten Fall eine Verbesserung der Genauigkeit von etwa 20% erreicht werden kann (vgl. Tabelle 4.2).
Tabelle 4.3 zeigt die Abweichung der Werte zu der „Nie Attachment“-Regel. Vergleicht man die beiden Tabellen, stellt man fest, dass die größte Steigerung beim Datensatz 3 erfolgt ist. Diese Steigerung hängt mit der Anzahl der Attachments im Datensatz

Accuracy (Abweichung zu Nie Attachment)	Datensatz 1	Datensatz 2	Datensatz 3
Einfacher Algorithmus	-7.28%	0.98%	3.68%
Erweiterter Algorithmus (2)	-4.85%	1.36%	2.21%
Paul Graham (0.51)	3.98%	11.53%	18.52%
Naive Bayes (0.5)	3.98%	10.96%	21.63%
Combined (2:1,0.34)	10.16%	12.09%	21.23%

Tabelle 4.3: Accuracy (Abweichung zu „Nie Attachment“) - Testdatensätze

zusammen. Datensatz 3 hat prozentual die größte Anzahl. Ein Grund dieser Steigerung könnte sein, dass die Algorithmen mehr Attachment-E-Mails gelernt haben, bevor sie die erste klassifizieren mussten.
Um nun unabhängige Ergebnisse für die Algorithmen zu bekommen, sind die Tests auf den beiden Kontrolldatensätzen durchgeführt worden. Die Accuracy-Werte überraschen auf den ersten Blick für den einfachen wie auch für den den erweiterten Algorithmus bei Datensatz 5. Sie liegen sehr hoch. Betrachtet man allerdings die „Nie Attachment“-Regel wird deutlich, dass diese schwächer sind (vgl. Tabelle 4.4).

Accuracy	Datensatz 4	Datensatz 5
Einfacher Algorithmus	65.96%	82.21%
Erweiterter Algorithmus (2)	68.79%	84.06%
Paul Graham (0.51)	84.40%	90.67%
Naive Bayes (0.5)	78.01%	90.78%
Combined (2:1,0.34)	95.04%	96.10%
Nie Attachment	75.18%	89.59%

Tabelle 4.4: Accuracy - Kontrolldatensätze

Diesen Sachverhalt macht Tabelle 4.5 noch klarer, wo die „Nie Attachment“-Werte abgezogen wurden.

Accuracy	Datensatz 4	Datensatz 5
Einfacher Algorithmus	-9.22%	-7.38%
Erweiterter Algorithmus (2)	-6.39%	-5.53%
Paul Graham (0.51)	9.22%	1.08%
Naive Bayes (0.5)	2.83%	1.19%
Combined (2:1,0.34)	19.86%	6.51%

Tabelle 4.5: Accuracy - (Abweichung zu „Nie Attachment“) - Kontrolldatensätze

Auch bei diesen Werten ist klar erkennbar, dass die höchste Verbesserung auf dem Datensatz mit dem größten Attachment-Anteil erzielt wurde. Datensatz 5 hat die geringste Erkennungssteigerungen im Vergleich zu allen anderen Datensätzen aufzuweisen, aber

hat eine Accuracy von 96%. Es ist davon auszugehen, dass es schwieriger ist zwischen 90% und 100% die Accuracy zu verbessern als bei Werten, die darunter liegen.
Bei allen Accuracy-Werten besteht das Problem, dass nicht klar erkennbar ist, wie die Verbesserung der Erkennung erzielt wurde. Es kann eine Steigerung erzielt werden, indem wenige Attachment erkannt werden und alle E-Mails ohne Attachment oder indem alle Attachments erkannt werden und dafür mehr E-Mails falsch eingestuft werden.

4.3.3 Precision/Recall

Da die Accuracy-Werte alleine betrachtet einige Details nicht preisgeben, sind hier die Ergebnisse mit den Precision- und Recall-Werten dargestellt. Abbildung 4.6 zeigt die Werte der Aufstellung von den Testdatensätzen. Der erste Prozentsatz entspricht dem Precision und der zweite dem Recall-Wert. Ein perfekter Wert, wäre wenn Precision- und Recall 100% hätten. Dies würde bedeuten, dass alle E-Mails richtig erkannt wurden.

Precision/Recall	Datensatz 1		Datensatz 2		Datensatz 3	
Einfacher Algorithmus	23.81%	20.27%	52.29%	50.44%	62.54%	27.60%
Erweiterter Algorithmus (2)	78.81%	21.05%	52.76%	59.29%	53.49%	50.27%
Paul Graham (0.51)	70.45%	41.89%	82.80%	68.14%	95.00%	59.70%
Naive Bayes (0.5)	62.50%	60.81%	75.00%	76.99%	85.28%	79.92%
Combined (2:1,0.34)	76.14%	90.54%	76.67%	81.42%	88.00%	75.14%

Tabelle 4.6: Precision/Recall – Testdatensätze

Der einfache Algorithmus hat wieder sehr schlechte Werte. Er hat nur bei Datensatz 2 mittelmäßige Werte. Diese Werten reichen fast an die Werte des Naive Bayes im schlechtesten Falle von Datensatz 1 ran. Dennoch sind diese Werte ein Indiz dafür, dass der Algorithmus nur in den seltensten Fällen funktioniert. Der erweiterte Algorithmus schneidet leicht besser ab und liegt immer über dem einfachen Algorithmus, kommt aber nicht an die Werte der komplexeren Verfahren heran. Naive Bayes und Paul Graham schneiden etwa gleich gut ab. Bei Paul Graham sind Genauigkeiten von 95% zu sehen, bei Naive Bayes ist der Erkennungsanteil der Attachments dagegen größer. Gewichtet man Precision gegen Recall, stellt sich Paul Graham als Gewinner heraus, bei umgekehrter Gewichtung Naive Bayes.
Der Combined-Algorithmus liefert gute Precision und Recall-Werte. Im Datensatz 3 jedoch ist der Combined-Algorithmus leicht schlechter als der Naive Bayes, ein ähnliches Ergebnis konnte man bereits bei den Accuracy-Werten beobachten. Diese Verschlechterung könnte daher rühren, dass Paul Graham schlechter vorhersagt und die Spitze des Naive Bayes nach unten zieht. Im Allgemeinen ist aber festzuhalten, dass Combined mit Abstand die beste Klassifizierung aufweist. Die Precision bzw. Recall-Werte bestätigen diesen Eindruck. Die Precision-Werte liegen zwischen 76% und 88% und die Recall-

Werte zwischen 75% und 90%, dadurch lässt sich auch im schlechtesten Fall ablesen, dass der Algorithmus gut funktioniert. Bemerkenswert ist hier, dass teilweise deutliche Schwächen von Naive Bayes und Paul Graham ausgeglichen werden. Im Fall, bei dem Naive Bayes einen Precision-Wert von 62.50% und einen Recall-Wert von 60.81% und der Paul Graham Algorithmus 70.45% bzw. 41.89% hat, erreicht der Combined 76.14% und 90.54%. Dieses Verhalten lässt sich damit erklären, dass Paul Graham und Naive Bayes andere E-Mails als anhangverdächtig klassifiziert. Wird bei einem der beiden Algorithmen mit hoher Wahrscheinlichkeit ein Attachment gefunden, wird dies auch gemeldet. Ist die Wahrscheinlichkeit aber grenzwertig, entscheidet der Algorithmus mit Hilfe des jeweils anderen Verfahren und somit kommt es zur Glättung der Ergebnisse. Dies ist sehr zum Vorteil von True Positive und False Positives.
Betrachtet man zusätzlich zu den Testdatensätzen die Kontrolldatensätze, fallen beide Datensätze mit sehr guten Ergebnissen beim Combined-Algorithmus auf (vgl. Tabelle 4.7). Datensatz 4 liefert über 97% aller Attachments bei einer Genauigkeit von 85%. Aber auch Datensatz 5 mit jeweils über 80% ist erstaunlich gut.

Precision/Recall	Datensatz 4		Datensatz 5	
Einfacher Algorithmus	25.93%	20.00%	27.92%	44.79%
Erweiterter Algorithmus (2)	37.14%	37.14%	33.33%	53.13%
Paul Graham (0.51)	82.50%	47.14%	69.23%	18.75%
Naive Bayes (0.5)	56.25%	51.43%	58.73%	38.54%
Combined (2:1,0.34)	85.00%	97.14%	84.91%	80.21%

Tabelle 4.7: Precision/Recall – Kontrolldatensätze

Zwei interessante Datensätze sind die Datensätze 2 und 4. Bei Datensatz 2 bewegen sich die Ergebnisse sehr nahe beieinander, was daran zu erkennen ist, dass die Linien des Precision-Recall-Diagramms ebenfalls sehr nahe beieinander liegen (vgl. Abbildung 4.12). Im Gegensatz zeichnen sich bei Datensatz 4 ab einem Recall-Wert von 0.5 deutliche Unterschiede zwischen den Algorithmen ab (vgl. Abbildung 4.13). Dies zeigt, dass die Algorithmen datensatzbedingt schlechter oder besser sein können.

4.3.4 ROC-Kurve/AUC

Die ROC-Kurven spiegeln das Bild der Ergebnisse des Information Retrievals wieder. Eine sehr gute Trennung ist mit dem kombinierten Algorithmus möglich. Die schlechteste Trennung wird mit den beiden einfachen Algorithmen erreicht.
In der Tabelle 4.8 ist gut zu erkennen, dass der Combined-Algorithmus immer über 0.95 liegt, was ein sehr gutes Ergebnis bedeutet. Paul Graham und Naive Bayes liegen immer unter dem Combined-Algorithmus, aber immer über den beiden einfachen Algorithmen. Der erweiterte Algorithmus hat im Datensatz 3 ein erstaunlich gutes Ergebnis von 0.81. Dieses Ergebnis ist auf die große Menge an E-Mails und im Aufbau sehr ähnlichen E-Mails mit Attachment zurückzuführen.

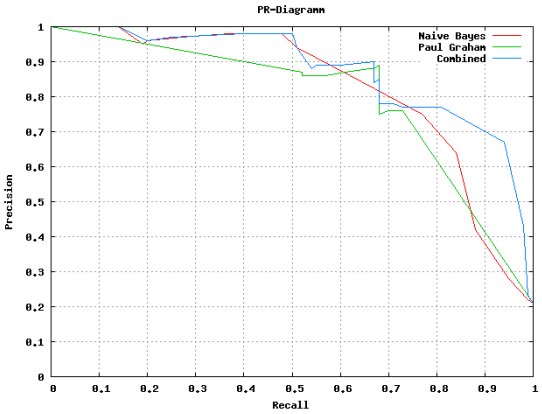

Abbildung 4.12: PR - Datensatz 2

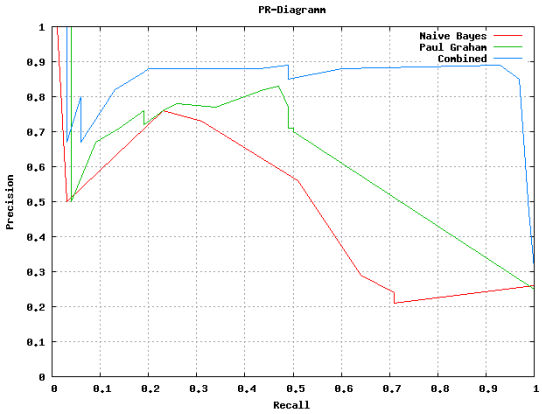

Abbildung 4.13: PR - Datensatz 4

AUC	Datensatz 1	Datensatz 2	Datensatz 3
Einfacher Algorithmus	0.54	0.69	0.60
Erweiterter Algorithmus	0.51	0.68	0.81
Paul Graham	0.90	0.86	0.90
Naive Bayes	0.88	0.90	0.94
Combined	0.97	0.95	0.97

Tabelle 4.8: AUC-Werte - Testdatensätze

Die Abbildung 4.14 veranschaulicht, wie der einfache und erweiterte Algorithmus im Vergleich zum Combined-Algorithmus im Datensatz 1 abschneidet.
Eingezeichnet ist in der Abbildung die Gerade mit der Steigung $m = 1$, die angibt, dass der Flächeninhalt 0.5 ist. Da die beiden Algorithmen nahe bei 0.5 liegen, sind die Algorithmen leicht darüber. Mit weitem Abstand ist der Combined zu erkennen. Dieser deckt mit einem AUC von 0.97 fast die komplette Fläche ab. Naive Bayes und Paul Graham sind etwa gleich gut, erkennbar daran, dass die Kurven nicht weit voneinander entfernt sind.

AUC	Datensatz 4	Datensatz 5
Einfacher Algorithmus	0.51	0.66
Erweiterter Algorithmus	0.58	0.70
Paul Graham	0.86	0.84
Naive Bayes	0.66	0.81
Combined	0.96	0.97

Tabelle 4.9: AUC-Werte - Kontrolldatensätze

Nimmt man zur Bewertung die Kontrolldatensätze hinzu, fällt wieder ein relativ guter AUC-Wert beim Datensatz 5 auf. Ein AUC von 0.70 liegt im Bereich des Akzeptablen,

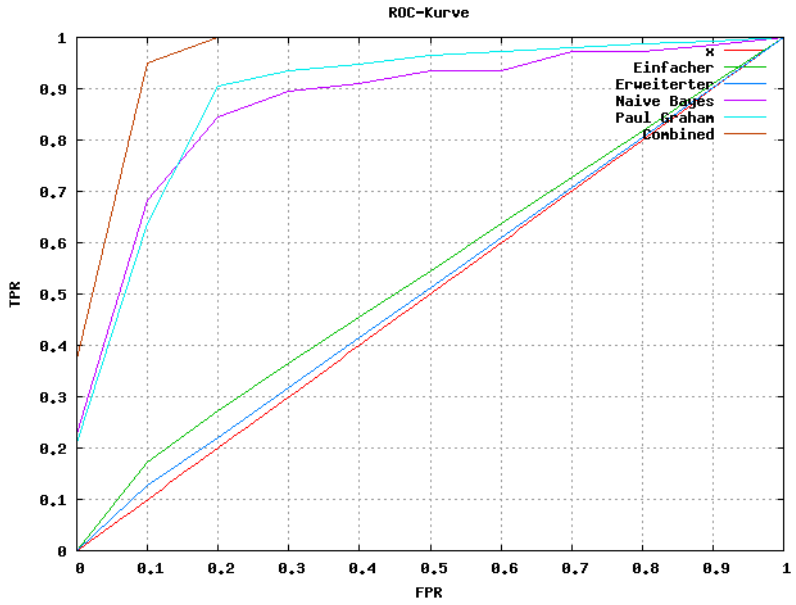

Abbildung 4.14: ROC-Kurve - Datensatz 1

der aber durch Datensatz 4 relativiert wird. Weiterhin fällt ein sehr schlechter AUC von Datensatz 4 auf. In der Abbildung 4.15 ist dies an der Schlangenlinie zu erkennen.

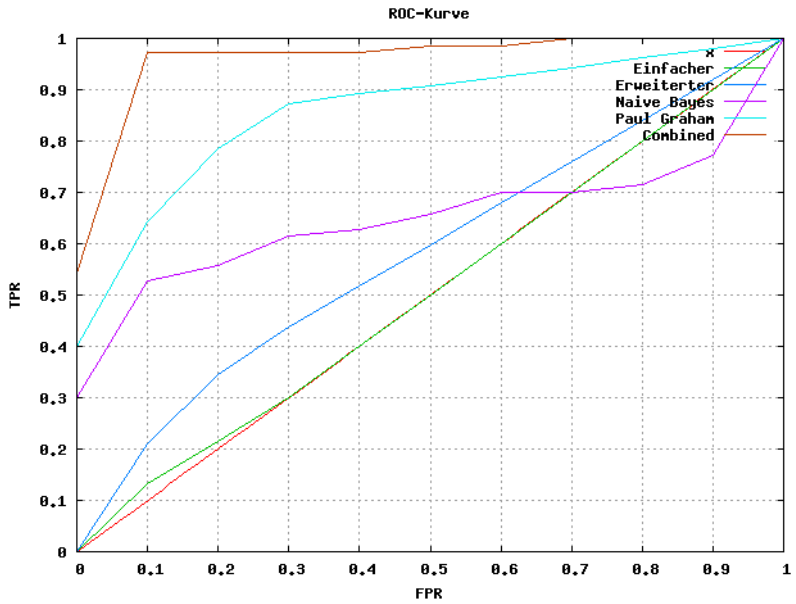

Abbildung 4.15: ROC-Kurve Datensatz 4

Eine Kurve dieser Form entsteht durch die Bildung der Durchschnittswerte. Einige Durchgänge der Cross-Validation müssen so schlechte Ergebnisse gebracht haben, dass

dies sich auch auf den Durchschnitt auswirkt. Abbildung 4.16 und Abbildung 4.17 zeigen zwei der Einzelkurven. Die Kurve, die im ersten Durchgang entstanden ist, liegt unter der Mittellinie, was ein sehr schlechtes Ergebnis ist. Man könnte nun annehmen, dass die Klassifizierung im Falle des ersten Durchgangs mit negierter Voraussage bessere Ergebnisse bringen könnte. Aber da Naive Bayes im gleichen Datensatz im zehnten Durchgang ein deutlich besseres Ergebnis erzielt, ist diese Annahme nicht korrekt. Weiterhin hat der Naive Bayes bei allen anderen Datensätze gute Ergebnisse erzielt.

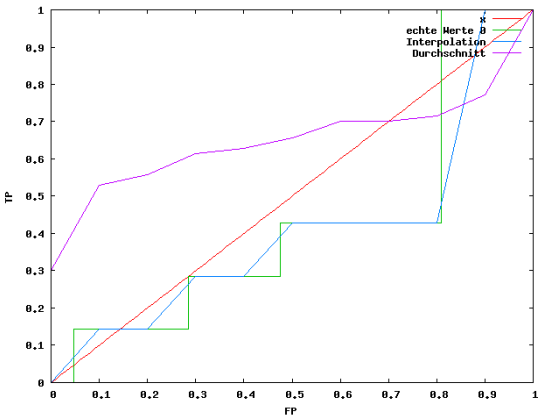

Abbildung 4.16: Datensatz 4 (1)

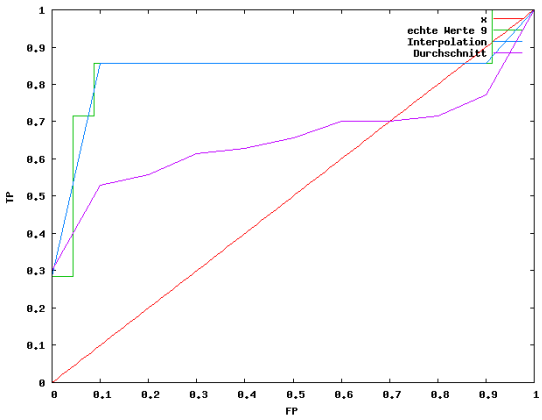

Abbildung 4.17: Datensatz 4 (10)

Trotz des schlechten Ergebnisses von Naive Bayes zeigt der Combined ein gutes Resultat. Der Naive Bayes scheint vom Combined soweit durch den Paul Graham vernachlässigt zu werden, dass nur die eindeutigen E-Mails gemeldet werden.
Tabelle 4.9 zeigt nochmals, dass auch bei unbekannten Datensätzen das Ergebnis nicht zu sehr von den Testdatensätzen abweicht.

4.3.5 Schlussfolgerung

Bei allen drei Maßen ist zu erkennen, dass der Combined-Algorithmus das beste Ergebnis hervorgebracht hat. Dicht gefolgt von den beiden Algorithmen Paul Graham und Naive Bayes. Mit Abstand die schlechtesten Algorithmen sind der einfache und der erweiterte Algorithmus.
Die Einzelanalysen haben in etwa das gleiche Ergebnis gebracht. Precision und Recall zeigen, dass eine sehr gute Erkennung möglich ist und die ROC-Kurve, dass die Trennung der Klassen prinzipiell durchführbar ist.
Da die Ergebnisse über wenige Datensätze erhoben wurden, können bei anderen Datensätzen, vor allem aus anderen Sprachen erhebliche Unterschiede resultieren.

4.3.6 Faktor bei erweiterten Algorithmus

Beim erweiterten Algorithmus ist mit Faktor auf zwei gewählt, da in den Tabellen zu erkennen ist, dass ein zu großer Faktor die Genauigkeit gegen die „Nie Attachment“-

Regel konvergieren lässt. Das ist damit zu erklären, dass der Faktor so groß wird, dass ab einem gewissen Zeitpunkt die E-Mailmenge nicht mehr genügend Worte umfasst, um einen so hohen Unterschied an Worthäufigkeiten abzubilden.
Die genauen Tabellen sind im Anhang zu finden.

4.3.7 Verhältnis bei Combined

Um die richtigen Verhältnisse beim Combined-Algorithmus herauszufinden, wurden die ROC-Werte bei verschiedenen Faktoren gemessen. Auf den Testdatensätzen sind die AUC-Werte der ROC-Kurve mit verschiedenen Verhältnissen gemessen worden.
Da Datensatz 1 und Datensatz 3 ein Verhältnis von zweimal Naive Bayes zu einmal Paul Graham als besten Wert haben, wurde dieser verwendet (vgl. Abbildung 4.18 und Abbildung 4.19).

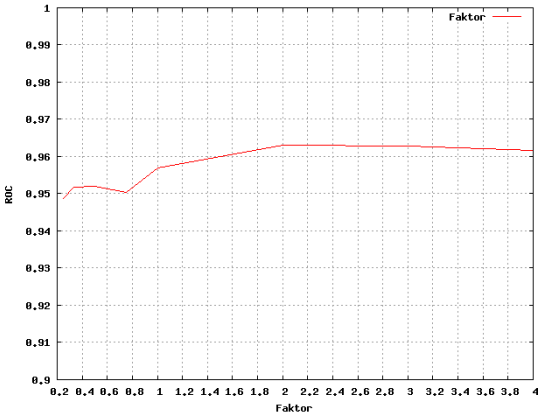

Abbildung 4.18: Combined Datensatz 1

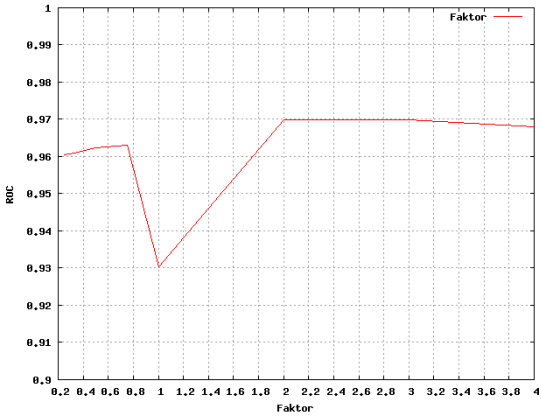

Abbildung 4.19: Combined Datensatz 3

Datensatz 2 würde eher ein Wert von viermal Naive Bayes zu einmal Paul Graham als besten Wert sehen.

4.3.8 Speicherverbrauch

Der erweiterte Algorithmus, Naive Bayes, Paul Graham und der Combined-Algorithmus benutzen Wortlisten um die E-Mails zu speichern. Da diese Wortlisten auf der Festplatte wie auch im Speicher während der Verarbeitung liegen, ist deren Größe ein interessanter Faktor.
Da aber die Optimierung des Speicherverbrauchs nicht im Vordergrund dieser Arbeit stand, wird hier lediglich eine Messung durchgeführt.
Im Deutschen gibt es zwischen 300.000 und 500.000 Worte und die englische Sprache umfasst ca. 600.000 bis 800.000 Worte. Aus diesem Grund wurden automatisch ca. 2.500.000 verschiedene Worte erzeugt, dies entspricht der doppelten Menge beider Sprachen zusammen. Abbildung 4.20 zeigt, dass die Steigung linear ist und die Maximalgröße ca. 50 MegaByte, was in heutiger Zeit vernachlässigbar ist.

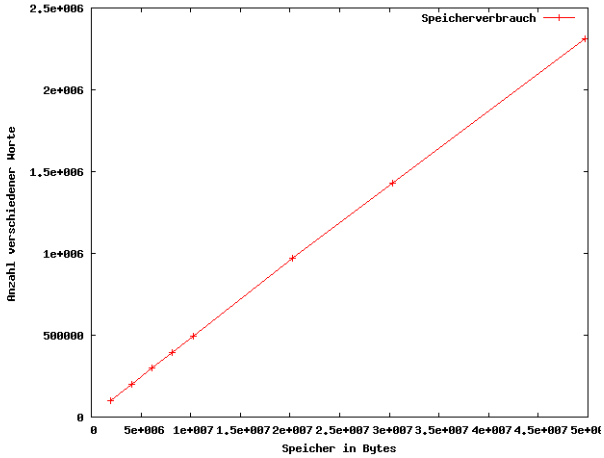

Abbildung 4.20: Speicherverbrauch in Abhängigkeit zur Wortanzahl

Kapitel 5

Fazit

5.1 Fazit

Der praktische und der schriftliche Teil der Arbeit haben gezeigt, dass es möglich ist, ein Plugin für Thunderbird zu entwickeln, dass anhangverdächtige E-Mails findet. Thunderbird ist in diesem Zusammenhang nur exemplarisch zu verstehen, da eine Portierung auf andere E-Mail-Anwendungen möglich ist (vgl. Abschnitt 5.2.2).
Da die verwendeten Techniken aus dem Spam-Problem abgeleitet sind, kamen auch nur Verfahren zum Einsatz, die dort wirksam sind. Der einfache Algorithmus, der einige Schwächen hat, zeigt auch in den Analysen keine guten Ergebnisse. Die Schwächen lassen sich in dem Beachten eines einziges Wortes und der manuellen Pflege der Wörter beziffern. Im Gegenzug ist dieses Verfahren sehr einfach für den Benutzer zu verstehen, was einen Bedienungsvorteil ausmacht. Ein Benutzer kann direkt einschätzen, aus welchem Grund eine bestimmte Klassifizierung stattgefunden hat.
Der erweiterte Algorithmus sollte durch ein Verfahren zum Finden der Wortmenge eine Verbesserung der Ergebnisse erzielen. Allerdings sind die Ergebnisse nicht wesentlich besser als die des einfachen Algorithmus ausgefallen. Nachteile sind das Verlorengehen des Bedienungsvorteils, sowie die Gefahr, dass der erweiterte Algorithmus je nach Einstellung des Verhältnisses keine Attachments mehr erkennen kann, da die Anzahl der gesendeten E-Mails zu klein ist.
Der Naive Bayes-Algorithmus ist ein sehr guter Klassifizierer in diesem Problem. Er nutzt alle Wörter in einer E-Mail zur Klassifizierung und lernt von den gesendeten E-Mails des Nutzers. Ebenso der Paul Graham Algorithmus. Die beiden Algorithmen unterscheiden sich dennoch grundlegend, da in einem Fall mit Wahrscheinlichkeiten und im anderen Fall mit Verhältnissen gerechnet wird.
Da beide Algorithmen gut sind, aber eine andere Menge an E-Mails abdecken, ist es sinnvoll einen kombinierten Algorithmus zu erstellen, der die Vorteile beider Algorithmen ausnutzt. Die Evaluierung hat gezeigt, dass die Kombination der Algorithmen die Erkennungsraten nochmals verbessert hat. Die Precision- und Recall-Werte sind sogar auf über 80% gestiegen.

Von einer perfekten Erkennung kann man hier noch nicht sprechen, allerdings ist dies erst der Anfang das Problem anhangverdächtiger E-Mails zu lösen. Dennoch ist das Ziel der Arbeit erreicht, da ein funktionsfähiges Plugin für Thunderbird entstanden ist. Dieses kann als Grundlage für weitere Forschungsarbeit in diesem Thema genutzt werden.

5.2 Ausblick

In diesem Abschnitt werden Verbesserungsvorschläge sowie weitere Funktionen vorgestellt, die den AttachmentChecker für den Nutzer angenehmer machen können.

5.2.1 JavaScript-Version

Da in der vorgestellten Version der Python-Interpreter immer benötigt wird, wäre eine JavaScript-Variante des Plugins eine Möglichkeit, die Verwendung des Attachment-Checkers auch auf anderen Plattformen zu ermöglichen. Diese Variante ist nur dann zu empfehlen, wenn keine weiteren Tests auf den Algorithmen durchgeführt werden müssen.

5.2.2 Portierung auf andere Plattformen

Interessant ist das Plugin auch für den Firmenbereich, wo die Anzahl der E-Mails deutlich höher als im Privatbereich ist. Da die meisten Firmen Microsoft Outlook einsetzen, müsste eine geeignete Implementierung geschaffen werden. Ein Möglichkeit wäre ebenfalls ein Plugin, dass mit Python kommuniziert. Dies hätte den Vorteil, dass die Wartung des Plugins keinen Mehraufwand bedeutet. Eventuell wäre in diesem Zusammenhang auch eine zentrale Verwaltung von Wortlisten sinnvoll. Dafür müssten aber weitere Tests bezüglich der Unterscheidung von Wortlisten pro Person gemacht werden.

5.2.3 Onlinevisualisierung der Klassifikation

Eine visuelle Darstellung, wie wahrscheinlich es ist, dass an die E-Mail ein Attachment sollte, wäre für den Benutzer interessant. Hier sind False Positiv-Meldungen nicht so störend wie wenn ein Popup erscheint. Abbildung 5.1 zeigt eine Möglichkeit. Der rote Balken zeigt die Wahrscheinlichkeit auf, dass ein Attachment an die E-Mail gehört. Ein grüner Balken wiederum, dass es unwahrscheinlich ist. Ein oranger Balken könnte anzeigen, dass die Klassifizierung nicht ganz eindeutig ist. Eine Veränderung des Plugins wäre notwendig, da zur Zeit keine Wahrscheinlichkeiten an das Plugin gegeben werden. Die Wahrscheinlichkeiten müssten dann direkt von Thunderbird ausgewertet werden. Außerdem muss auf die Laufzeit der Klassifizierung geachtet werden. Dauert eine Klassifizierung zu lange, ist die Farbe des Balkens auf einem alten Stand und verwirrt den

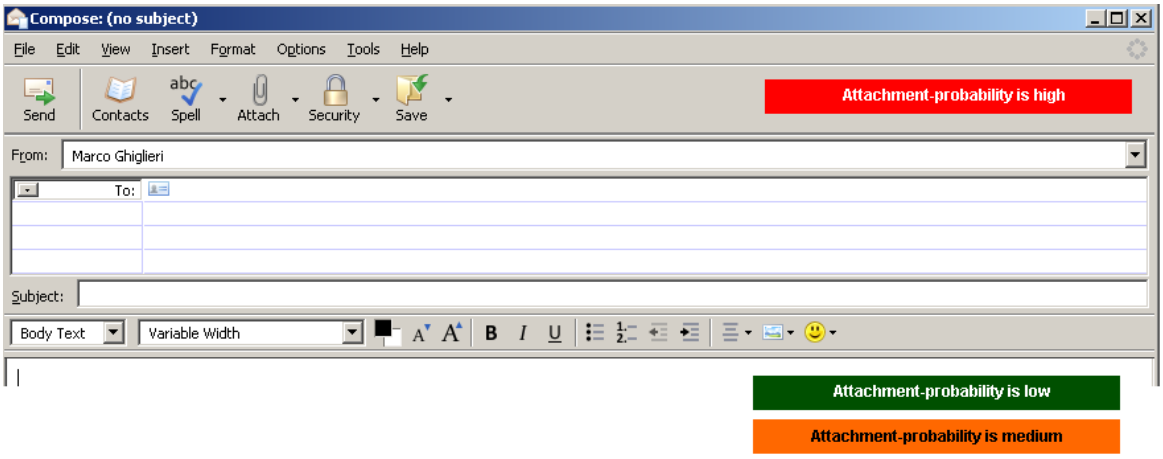

Abbildung 5.1: Erweiterung Farbbalken

Benutzer. Eine Schätzung des Ergebnisses der Wahrscheinlichkeit wäre auch denkbar. Hat man beispielsweise gerade eine Wahrscheinlichkeit von 0.99, ist es unwahrscheinlich, dass sie beim nächsten Wort auf 0.01 fällt, somit ist eine Prüfung nach jedem fünften Wort oder nach Ablauf einer bestimmten Zeit ebenfalls praktikabel. Ob dann das Popup bei einem Sendeversuch noch angezeigt wird, kann dem Benutzer überlassen werden.

5.2.4 Verbesserung des Lerners

In dieser Arbeit wurden Geschwindigkeitsmaßnahmen vernachlässigt. Es konnte im Laufe der Arbeit allerdings keine Engpässe festgestellt werden. Die durchaus langsamen Lernzeiten beim manuellen Lernen sind auf einen IMAP-Server zurückzuführen. Bei lokal gespeicherten E-Mails ist die Verarbeitung wesentlich schneller.
Die Zugriffszeiten auf die Datenbank sind wesentlich entscheidender für den Nutzer. Die Zugriffszeiten nehmen vermutlich mit der Anzahl der Wörter zu, deswegen ist eine Optimierung, zum Beispiel die Wortlisten durch geeignete Verfahren des maschinellen Lernens zu verkleinern, sinnvoll. Dadurch verkürzen sich nicht nur die Zugriffszeiten auf einzelne Worte, sondern auch der Speicherplatz, der benötigt wird um die Datenbank abzulegen, verringert sich.

5.2.5 Automatisches Lernen

Ein im Hintergrund arbeitender Lernprozess könnte das anfängliche Lernen des AttachmentCheckers ersetzen. Der Vorteil für den Nutzer wäre, dass der AttachmentChecker bei Updates oder Neuinstallation nicht manuell gelernt werden muss. Problem, dass sich ergibt ist, dass man eine Liste führen muss, die aufzeichnet, welche E-Mails bereits gelernt wurden, da man damit rechnen muss, dass der Nutzer Thunderbird jederzeit schließt und damit das Lernen abbricht.

5.2.6 Reaktion des Nutzers

Da es sehr oft der Fall ist, dass man nach einem vergessenen Attachment, sofort eine E-Mail mit dem Attachment schickt, könnte der AttachmentChecker diese Reaktion analysieren und die Worte dementsprechend stärker gewichten. Dies hätte den Vorteil, dass E-Mails, die einen solchen Aufbau haben, nicht noch einmal ohne Hinweis gesendet werden. Realisierbar ist dies mit einer höheren Gewichtung der beim zweiten Mal gesendeten E-Mail. Wie die Erkennung dieser E-Mail abläuft, bleibt eine offene Frage.

5.2.7 Dynamische Anpassung des Schwellwerts

Aus algorithmischer Sicht müsste man einen Ansatz finden, wie man den Benutzerschwellwert so setzt, dass ein optimales Ergebnis des Algorithmus erzielt wird. Im Combined-Algorithmus schwanken die guten Erkennungsschwellwerte zwischen 0.33 und 0.35, welches einen Mittelwert von 0.34 ergibt, der auch als Standard nun eingepflegt ist. Probleme kann es geben, wenn der Benutzer einen Schwellwert von 0.4 bräuchte. Eventuell kann der Weg über die bereits klassifizierten E-Mails gemacht werden. Die Kalibrierung muss permanent angepasst werden. Kommt eine neue E-Mail mit Attachment wird der Schwellwert so angepasst, dass diese E-Mail gerade noch enthalten ist. Vorsicht ist geboten, wenn eine E-Mail zu extrem außerhalb der Grenzen liegt, dann sollte diese nicht beachtet werden oder eine leichte Korrektur des Schwellwertes stattfinden.

5.2.8 Auswahl des besten Algorithmus

Im AttachmentChecker sind derzeit fünf Algorithmen implementiert. Je nach Anwendungsfall kann es sinnvoll sein einen anderen als den Combined-Algorithmus zu benutzen. Um dies automatisch durchführen zu können, könnte man zu Beginn auf der Datenmenge des Nutzers jeden Algorithmus testen und den besten benutzen. Ein offenes Problem ist, ob mit der Zeit nicht ein anderer Algorithmus bessere Ergebnisse liefern könnte.

5.2.9 Abhängige Worte

In den derzeitigen Algorithmen wird jedes Wort unabhängig betrachtet. Einen Vorteil könnte es bringen, Worte abhängig zu betrachten. Beispielsweise deutet das Wort „Anhang“ auf ein Attachment hin, wohingegen die beiden Wörter „kein Anhang“ in Kombination auf kein Attachment hinweisen. Untersucht werden müsste, ob sich der Mehraufwand für die Analyse von Abhängigkeiten unter Worten im Vergleich zur Laufzeit und zum Speicherverbrauch lohnen. Der Speicherverbrauch steigt rapide an. Betrachtet man immer nur zwei Worte als abhängig, erhöht sich bei einer Wortmenge von zehn Worten der Speicherverbrauch von zehn Speichereinheiten auf 19 Speichereinheiten, was fast eine Verdoppelung entspricht. Mit geeigneten Kompressionsverfahren

und Vorauswahlen müsste dem entgegengewirkt werden können.

5.3 Rückblick

Begonnen hat die Arbeit mit der fraglichen Aussicht, ob es überhaupt möglich ist, E-Mails mit Attachments anhand der charakteristischen Merkmale des E-Mail-Textes zu erkennen. Die Erkennung wurde vorerst zurückgestellt und der einfache Algorithmus implementiert. Mit dem einfachen Algorithmus war es recht einfach möglich, das Plugin zu entwickeln und zu testen. Nachdem die Aussicht auf ein laufendes Plugin gut war, kam der erste Algorithmus hinzu. Es war eine Form des Naive Bayes, der aber durch den Unterlauf keine guten Werte brachte. Ohne darauf einzugehen, implementierte ich den Paul Graham Algorithmus, der direkt gute Ergebnisse brachte. Ein weiterer Versuch den Naive Bayes lauffähig zu machen, löste dann das Unterlauf-Problem. Der erweiterte Algorithmus sollte das Problem der Wortmenge lösen, da die erste Annahme war, dass der einfache Algorithmus die besten Ergebnisse erzielt. Doch sowohl der einfache als auch der erweiterte Algorithmus erzielen im Endeffekt die schlechtesten Ergebnisse.
Der Combined-Algorithmus entstand aus der Feststellung, dass Naive Bayes und Paul Graham verschiedene E-Mails als anhangverdächtig identifizieren.
Ein schwerwiegendes Problem war und ist das Auffinden von geeigneten Testdatensätzen. Das Problem liegt darin, dass es nicht möglich ist E-Mails so unkenntlich zu machen, dass jegliche Analysen nicht verfälscht werden. Eigene E-Mails stellt keiner gerne der Öffentlichkeit zur Verfügung. Ein Anfang wären aber sicherlich gehashte Wortlisten, die kein Rückschluss mehr auf die Originalemails zulassen.
Während der Entwicklung des Plugins ist außerdem aufgefallen, dass die Dokumentation von Mozilla verbesserungsbedüftig ist. Durch eine Verbesserung können die Einstiegshürden in die Erweiterung von Thunderbird erniedrigt werden. Die schwere Auffindbarkeit von Informationen hat das Erstellen des Plugins erschwert.
Entstanden ist dennoch ein lauffähiger AttachmentChecker, der hoffentlich viele Attachments identifiziert. Dennoch ist der AttachmentChecker bei weitem nicht fertiggestellt, da sich immer wieder neue Stellen eröffnen, die eine Weiterentwicklung ermöglichen.
Diese Arbeit war sehr interessant. Vor allem die Erkenntnis, dass mit dem Ergebnis quasi jedem Nutzer bei einem Alltagsproblem geholfen werden kann, erfreut. Das schönste ist, dass die Arbeit mit nur einem Satz „Unwissenden“ zu erklären ist: „Eine Erweiterung für ein E-Mail-Programm, dass erkennt, ob Sie einen Anhang vergessen haben!“. Jeder versteht, was es tun soll, aber wenige wissen, wie es zu lösen ist.

Kapitel 6

Anhang

6.1 Ergebnistabellen

6.1.1 Datensatz 1

6.1.1.1 Einfacher Algorithmus

para	TP	TN	FP	FN	Total	Total Attachments	Accuracy	Precision	Recall	Fallout
1	15	331	48	59	453	74	76,38%	23,81%	20,27%	12,66%

6.1.1.2 Erweiterter Algorithmus

para	TP	TN	FP	FN	Total	Total Attachments	Accuracy	Precision	Recall	Fallout
2	8	349	30	66	453	74	78,81%	21,05%	10,81%	7,92%
3	1	361	18	73	453	74	79,91%	5,26%	1,35%	4,75%
4	3	365	14	71	453	74	81,24%	17,65%	4,05%	3,69%
5	1	367	12	73	453	74	81,24%	7,69%	1,35%	3,17%
6	1	373	6	73	453	74	82,56%	14,29%	1,35%	1,58%
7	0	378	1	74	453	74	83,44%	0,00%	0,00%	0,26%
8	0	376	3	74	453	74	83,00%	0,00%	0,00%	0,79%
9	0	378	1	74	453	74	83,44%	0,00%	0,00%	0,26%
10	0	378	1	74	453	74	83,44%	0,00%	0,00%	0,26%
11	0	378	1	74	453	74	83,44%	0,00%	0,00%	0,26%
12	0	378	1	74	453	74	83,44%	0,00%	0,00%	0,26%
13	0	378	1	74	453	74	83,44%	0,00%	0,00%	0,26%
14	0	378	1	74	453	74	83,44%	0,00%	0,00%	0,26%
15	0	378	1	74	453	74	83,44%	0,00%	0,00%	0,26%

6.1.1.3 Paul Graham

para	TP	TN	FP	FN	Total	Total Attachments	Accuracy	Precision	Recall	Fallout
0.01	38	354	25	36	453	74	86,53%	60,32%	51,35%	6,60%
0.02	38	354	25	36	453	74	86,53%	60,32%	51,35%	6,60%
0.03	37	354	25	37	453	74	86,31%	59,68%	50,00%	6,60%
0.04	35	354	25	39	453	74	85,87%	58,33%	47,30%	6,60%
0.05	35	354	25	39	453	74	85,87%	58,33%	47,30%	6,60%
0.06	34	354	25	40	453	74	85,65%	57,63%	45,95%	6,60%
0.07	34	354	25	40	453	74	85,65%	57,63%	45,95%	6,60%
0.08	34	354	25	40	453	74	85,65%	57,63%	45,95%	6,60%
0.09	34	354	25	40	453	74	85,65%	57,63%	45,95%	6,60%
0.1	34	354	25	40	453	74	85,65%	57,63%	45,95%	6,60%
0.11	33	354	25	41	453	74	85,43%	56,90%	44,59%	6,60%
0.12	33	354	25	41	453	74	85,43%	56,90%	44,59%	6,60%
0.13	33	354	25	41	453	74	85,43%	56,90%	44,59%	6,60%
0.14	33	354	25	41	453	74	85,43%	56,90%	44,59%	6,60%
0.15	32	354	25	42	453	74	85,21%	56,14%	43,24%	6,60%
0.16	32	355	24	42	453	74	85,43%	57,14%	43,24%	6,33%
0.17	32	355	24	42	453	74	85,43%	57,14%	43,24%	6,33%
0.18	31	355	24	43	453	74	85,21%	56,36%	41,89%	6,33%
0.19	31	355	24	43	453	74	85,21%	56,36%	41,89%	6,33%
0.2	31	356	23	43	453	74	85,43%	57,41%	41,89%	6,07%
0.21	31	356	23	43	453	74	85,43%	57,41%	41,89%	6,07%
0.22	31	356	23	43	453	74	85,43%	57,41%	41,89%	6,07%
0.23	31	356	23	43	453	74	85,43%	57,41%	41,89%	6,07%
0.24	31	356	23	43	453	74	85,43%	57,41%	41,89%	6,07%
0.25	31	356	23	43	453	74	85,43%	57,41%	41,89%	6,07%
0.26	31	356	23	43	453	74	85,43%	57,41%	41,89%	6,07%
0.27	31	356	23	43	453	74	85,43%	57,41%	41,89%	6,07%
0.28	31	356	23	43	453	74	85,43%	57,41%	41,89%	6,07%
0.29	31	356	23	43	453	74	85,43%	57,41%	41,89%	6,07%
0.3	31	356	23	43	453	74	85,43%	57,41%	41,89%	6,07%
0.31	31	356	23	43	453	74	85,43%	57,41%	41,89%	6,07%
0.32	31	357	22	43	453	74	85,65%	58,49%	41,89%	5,80%
0.33	31	357	22	43	453	74	85,65%	58,49%	41,89%	5,80%
0.34	31	357	22	43	453	74	85,65%	58,49%	41,89%	5,80%
0.35	31	357	22	43	453	74	85,65%	58,49%	41,89%	5,80%
0.36	31	357	22	43	453	74	85,65%	58,49%	41,89%	5,80%
0.37	31	357	22	43	453	74	85,65%	58,49%	41,89%	5,80%
0.38	31	357	22	43	453	74	85,65%	58,49%	41,89%	5,80%
0.39	31	357	22	43	453	74	85,65%	58,49%	41,89%	5,80%
0.4	31	357	22	43	453	74	85,65%	58,49%	41,89%	5,80%
0.41	31	357	22	43	453	74	85,65%	58,49%	41,89%	5,80%
0.42	31	357	22	43	453	74	85,65%	58,49%	41,89%	5,80%
0.43	31	357	22	43	453	74	85,65%	58,49%	41,89%	5,80%
0.44	31	357	22	43	453	74	85,65%	58,49%	41,89%	5,80%
0.45	31	357	22	43	453	74	85,65%	58,49%	41,89%	5,80%
0.46	31	357	22	43	453	74	85,65%	58,49%	41,89%	5,80%
0.47	31	357	22	43	453	74	85,65%	58,49%	41,89%	5,80%
0.48	31	357	22	43	453	74	85,65%	58,49%	41,89%	5,80%
0.49	31	357	22	43	453	74	85,65%	58,49%	41,89%	5,80%
0.5	31	357	22	43	453	74	85,65%	58,49%	41,89%	5,80%
0.51	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%
0.52	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%
0.53	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%
0.54	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%
0.55	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%
0.56	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%
0.57	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%
0.58	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%
0.59	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%

para	TP	TN	FP	FN	Total	Total Attachments	Accuracy	Precision	Recall	Fallout
0.6	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%
0.61	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%
0.62	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%
0.63	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%
0.64	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%
0.65	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%
0.66	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%
0.67	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%
0.68	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%
0.69	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%
0.7	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%
0.71	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%
0.72	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%
0.73	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%
0.74	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%
0.75	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%
0.76	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%
0.77	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%
0.78	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%
0.79	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%
0.8	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%
0.81	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%
0.82	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%
0.83	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%
0.84	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%
0.85	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%
0.86	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%
0.87	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%
0.88	30	366	13	44	453	74	87,42%	69,77%	40,54%	3,43%
0.89	16	373	6	58	453	74	85,87%	72,73%	21,62%	1,58%
0.9	8	376	3	66	453	74	84,77%	72,73%	10,81%	0,79%
0.91	8	376	3	66	453	74	84,77%	72,73%	10,81%	0,79%
0.92	8	376	3	66	453	74	84,77%	72,73%	10,81%	0,79%
0.93	8	376	3	66	453	74	84,77%	72,73%	10,81%	0,79%
0.94	7	376	3	67	453	74	84,55%	70,00%	9,46%	0,79%
0.95	7	376	3	67	453	74	84,55%	70,00%	9,46%	0,79%
0.96	7	376	3	67	453	74	84,55%	70,00%	9,46%	0,79%
0.97	5	376	3	69	453	74	84,11%	62,50%	6,76%	0,79%
0.98	5	376	3	69	453	74	84,11%	62,50%	6,76%	0,79%
0.99	5	376	3	69	453	74	84,11%	62,50%	6,76%	0,79%
1.00	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%

6.1.1.4 Naive Bayes

para	TP	TN	FP	FN	Total	Total Attachments	Accuracy	Precision	Recall	Fallout
0.0	74	0	379	0	453	74	16,34%	16,34%	100,00%	100,00%
0.01	74	0	379	0	453	74	16,34%	16,34%	100,00%	100,00%
0.02	74	2	377	0	453	74	16,78%	16,41%	100,00%	99,47%
0.03	74	9	370	0	453	74	18,32%	16,67%	100,00%	97,63%
0.04	74	9	370	0	453	74	18,32%	16,67%	100,00%	97,63%
0.05	74	9	370	0	453	74	18,32%	16,67%	100,00%	97,63%
0.06	74	9	370	0	453	74	18,32%	16,67%	100,00%	97,63%
0.07	74	9	370	0	453	74	18,32%	16,67%	100,00%	97,63%
0.08	74	9	370	0	453	74	18,32%	16,67%	100,00%	97,63%
0.09	74	9	370	0	453	74	18,32%	16,67%	100,00%	97,63%
0.1	74	9	370	0	453	74	18,32%	16,67%	100,00%	97,63%
0.11	74	9	370	0	453	74	18,32%	16,67%	100,00%	97,63%
0.12	74	9	370	0	453	74	18,32%	16,67%	100,00%	97,63%
0.13	74	9	370	0	453	74	18,32%	16,67%	100,00%	97,63%
0.14	74	9	370	0	453	74	18,32%	16,67%	100,00%	97,63%
0.15	74	9	370	0	453	74	18,32%	16,67%	100,00%	97,63%
0.16	74	9	370	0	453	74	18,32%	16,67%	100,00%	97,63%
0.17	74	9	370	0	453	74	18,32%	16,67%	100,00%	97,63%
0.18	74	9	370	0	453	74	18,32%	16,67%	100,00%	97,63%
0.19	74	9	370	0	453	74	18,32%	16,67%	100,00%	97,63%
0.2	74	9	370	0	453	74	18,32%	16,67%	100,00%	97,63%
0.21	74	9	370	0	453	74	18,32%	16,67%	100,00%	97,63%
0.22	74	9	370	0	453	74	18,32%	16,67%	100,00%	97,63%
0.23	74	9	370	0	453	74	18,32%	16,67%	100,00%	97,63%
0.24	74	9	370	0	453	74	18,32%	16,67%	100,00%	97,63%
0.25	74	9	370	0	453	74	18,32%	16,67%	100,00%	97,63%
0.26	74	9	370	0	453	74	18,32%	16,67%	100,00%	97,63%
0.27	74	9	370	0	453	74	18,32%	16,67%	100,00%	97,63%
0.28	74	9	370	0	453	74	18,32%	16,67%	100,00%	97,63%
0.29	74	9	370	0	453	74	18,32%	16,67%	100,00%	97,63%
0.3	74	9	370	0	453	74	18,32%	16,67%	100,00%	97,63%
0.31	74	9	370	0	453	74	18,32%	16,67%	100,00%	97,63%
0.32	74	9	370	0	453	74	18,32%	16,67%	100,00%	97,63%
0.33	74	9	370	0	453	74	18,32%	16,67%	100,00%	97,63%
0.34	74	9	370	0	453	74	18,32%	16,67%	100,00%	97,63%
0.35	74	9	370	0	453	74	18,32%	16,67%	100,00%	97,63%
0.36	74	9	370	0	453	74	18,32%	16,67%	100,00%	97,63%
0.37	74	9	370	0	453	74	18,32%	16,67%	100,00%	97,63%
0.38	74	9	370	0	453	74	18,32%	16,67%	100,00%	97,63%
0.39	74	9	370	0	453	74	18,32%	16,67%	100,00%	97,63%
0.4	74	10	369	0	453	74	18,54%	16,70%	100,00%	97,36%
0.41	74	10	369	0	453	74	18,54%	16,70%	100,00%	97,36%
0.42	74	10	369	0	453	74	18,54%	16,70%	100,00%	97,36%
0.43	74	10	369	0	453	74	18,54%	16,70%	100,00%	97,36%
0.44	74	10	369	0	453	74	18,54%	16,70%	100,00%	97,36%
0.45	74	21	358	0	453	74	20,97%	17,13%	100,00%	94,46%
0.46	74	46	333	0	453	74	26,49%	18,18%	100,00%	87,86%
0.47	71	120	259	3	453	74	42,16%	21,52%	95,95%	68,34%
0.48	67	254	125	7	453	74	70,86%	34,90%	90,54%	32,98%
0.49	58	329	50	16	453	74	85,43%	53,70%	78,38%	13,19%
0.5	45	352	27	29	453	74	87,64%	62,50%	60,81%	7,12%
0.51	40	359	20	34	453	74	88,08%	66,67%	54,05%	5,28%
0.52	12	370	9	62	453	74	84,33%	57,14%	16,22%	2,37%
0.53	7	374	5	67	453	74	84,11%	58,33%	9,46%	1,32%
0.54	6	375	4	68	453	74	84,11%	60,00%	8,11%	1,06%
0.55	3	378	1	71	453	74	84,11%	75,00%	4,05%	0,26%
0.56	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.57	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.58	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.59	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%

para	TP	TN	FP	FN	Total	Total Attachments	Accuracy	Precision	Recall	Fallout
0.6	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.61	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.62	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.63	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.64	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.65	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.66	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.67	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.68	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.69	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.7	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.71	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.72	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.73	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.74	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.75	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.76	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.77	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.78	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.79	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.8	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.81	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.82	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.83	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.84	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.85	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.86	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.87	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.88	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.89	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.9	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.91	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.92	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.93	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.94	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.95	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.96	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.97	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.98	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.99	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
1.00	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%

6.1.1.5 Combined

para	TP	TN	FP	FN	Total	Total Attachments	Accuracy	Precision	Recall	Fallout
0.0	74	0	379	0	453	74	16,34%	16,34%	100,00%	100,00%
0.01	74	0	379	0	453	74	16,34%	16,34%	100,00%	100,00%
0.02	74	0	379	0	453	74	16,34%	16,34%	100,00%	100,00%
0.03	74	0	379	0	453	74	16,34%	16,34%	100,00%	100,00%
0.04	74	0	379	0	453	74	16,34%	16,34%	100,00%	100,00%
0.05	74	0	379	0	453	74	16,34%	16,34%	100,00%	100,00%
0.06	74	0	379	0	453	74	16,34%	16,34%	100,00%	100,00%
0.07	74	0	379	0	453	74	16,34%	16,34%	100,00%	100,00%
0.08	74	0	379	0	453	74	16,34%	16,34%	100,00%	100,00%
0.09	74	0	379	0	453	74	16,34%	16,34%	100,00%	100,00%
0.1	74	0	379	0	453	74	16,34%	16,34%	100,00%	100,00%
0.11	74	0	379	0	453	74	16,34%	16,34%	100,00%	100,00%
0.12	74	0	379	0	453	74	16,34%	16,34%	100,00%	100,00%
0.13	74	0	379	0	453	74	16,34%	16,34%	100,00%	100,00%
0.14	74	0	379	0	453	74	16,34%	16,34%	100,00%	100,00%
0.15	74	0	379	0	453	74	16,34%	16,34%	100,00%	100,00%
0.16	74	0	379	0	453	74	16,34%	16,34%	100,00%	100,00%
0.17	74	0	379	0	453	74	16,34%	16,34%	100,00%	100,00%
0.18	74	2	377	0	453	74	16,78%	16,41%	100,00%	99,47%
0.19	74	9	370	0	453	74	18,32%	16,67%	100,00%	97,63%
0.2	74	9	370	0	453	74	18,32%	16,67%	100,00%	97,63%
0.21	74	9	370	0	453	74	18,32%	16,67%	100,00%	97,63%
0.22	74	9	370	0	453	74	18,32%	16,67%	100,00%	97,63%
0.23	74	9	370	0	453	74	18,32%	16,67%	100,00%	97,63%
0.24	74	9	370	0	453	74	18,32%	16,67%	100,00%	97,63%
0.25	74	9	370	0	453	74	18,32%	16,67%	100,00%	97,63%
0.26	74	9	370	0	453	74	18,32%	16,67%	100,00%	97,63%
0.27	74	9	370	0	453	74	18,32%	16,67%	100,00%	97,63%
0.28	74	9	370	0	453	74	18,32%	16,67%	100,00%	97,63%
0.29	74	9	370	0	453	74	18,32%	16,67%	100,00%	97,63%
0.3	74	18	361	0	453	74	20,31%	17,01%	100,00%	95,25%
0.31	74	69	310	0	453	74	31,57%	19,27%	100,00%	81,79%
0.32	74	240	139	0	453	74	69,32%	34,74%	100,00%	36,68%
0.33	74	343	36	0	453	74	92,05%	67,27%	100,00%	9,50%
0.34	67	358	21	7	453	74	93,82%	76,14%	90,54%	5,54%
0.35	52	362	17	22	453	74	91,39%	75,36%	70,27%	4,49%
0.36	39	364	15	35	453	74	88,96%	72,22%	52,70%	3,96%
0.37	35	365	14	39	453	74	88,30%	71,43%	47,30%	3,69%
0.38	34	365	14	40	453	74	88,08%	70,83%	45,95%	3,69%
0.39	34	365	14	40	453	74	88,08%	70,83%	45,95%	3,69%
0.4	34	365	14	40	453	74	88,08%	70,83%	45,95%	3,69%
0.41	34	365	14	40	453	74	88,08%	70,83%	45,95%	3,69%
0.42	33	365	14	41	453	74	87,86%	70,21%	44,59%	3,69%
0.43	33	365	14	41	453	74	87,86%	70,21%	44,59%	3,69%
0.44	32	365	14	42	453	74	87,64%	69,57%	43,24%	3,69%
0.45	32	365	14	42	453	74	87,64%	69,57%	43,24%	3,69%
0.46	32	366	13	42	453	74	87,86%	71,11%	43,24%	3,43%
0.47	32	366	13	42	453	74	87,86%	71,11%	43,24%	3,43%
0.48	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%
0.49	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%
0.5	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%
0.51	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%
0.52	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%
0.53	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%
0.54	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%
0.55	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%
0.56	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%
0.57	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%
0.58	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%
0.59	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%

para	TP	TN	FP	FN	Total	Total Attachments	Accuracy	Precision	Recall	Fallout
0.6	31	366	13	43	453	74	87,64%	70,45%	41,89%	3,43%
0.61	30	366	13	44	453	74	87,42%	69,77%	40,54%	3,43%
0.62	30	366	13	44	453	74	87,42%	69,77%	40,54%	3,43%
0.63	30	366	13	44	453	74	87,42%	69,77%	40,54%	3,43%
0.64	18	368	11	56	453	74	85,21%	62,07%	24,32%	2,90%
0.65	7	376	3	67	453	74	84,55%	70,00%	9,46%	0,79%
0.66	7	376	3	67	453	74	84,55%	70,00%	9,46%	0,79%
0.67	6	376	3	68	453	74	84,33%	66,67%	8,11%	0,79%
0.68	5	376	3	69	453	74	84,11%	62,50%	6,76%	0,79%
0.69	5	376	3	69	453	74	84,11%	62,50%	6,76%	0,79%
0.7	4	378	1	70	453	74	84,33%	80,00%	5,41%	0,26%
0.71	1	378	1	73	453	74	83,66%	50,00%	1,35%	0,26%
0.72	0	378	1	74	453	74	83,44%	0,00%	0,00%	0,26%
0.73	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.74	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.75	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.76	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.77	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.78	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.79	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.8	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.81	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.82	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.83	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.84	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.85	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.86	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.87	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.88	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.89	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.9	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.91	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.92	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.93	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.94	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.95	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.96	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.97	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.98	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
0.99	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%
1.00	0	379	0	74	453	74	83,66%	100,00%	0,00%	0,00%

6.1.2 Datensatz 2

6.1.2.1 Einfacher Algorithmus

para	TP	TN	FP	FN	Total	Total Attachments	Accuracy	Precision	Recall	Fallout
1	57	366	52	56	531	113	79,66%	52,29%	50,44%	12,44%

6.1.2.2 Erweiterter Algorithmus

para	TP	TN	FP	FN	Total	Total Attachments	Accuracy	Precision	Recall	Fallout
2	67	321	97	46	531	113	73,07%	40,85%	59,29%	23,21%
3	67	358	60	46	531	113	80,04%	52,76%	59,29%	14,35%
4	66	370	48	47	531	113	82,11%	57,89%	58,41%	11,48%
5	60	371	47	53	531	113	81,17%	56,07%	53,10%	11,24%
6	57	387	31	56	531	113	83,62%	64,77%	50,44%	7,42%
7	52	384	34	61	531	113	82,11%	60,47%	46,02%	8,13%
8	49	388	30	64	531	113	82,30%	62,03%	43,36%	7,18%
9	42	393	25	71	531	113	81,92%	62,69%	37,17%	5,98%
10	42	394	24	71	531	113	82,11%	63,64%	37,17%	5,74%
11	39	401	17	74	531	113	82,86%	69,64%	34,51%	4,07%
12	42	399	19	71	531	113	83,05%	68,85%	37,17%	4,55%
13	37	409	9	76	531	113	83,99%	80,43%	32,74%	2,15%
14	38	408	10	75	531	113	83,99%	79,17%	33,63%	2,39%
15	38	408	10	75	531	113	83,99%	79,17%	33,63%	2,39%

6.1.2.3 Paul Graham

para	TP	TN	FP	FN	Total	Total Attachments	Accuracy	Precision	Recall	Fallout
0.0	113	0	418	0	531	113	21,28%	21,28%	100,00%	100,00%
0.01	82	392	26	31	531	113	89,27%	75,93%	72,57%	6,22%
0.02	81	392	26	32	531	113	89,08%	75,70%	71,68%	6,22%
0.03	81	392	26	32	531	113	89,08%	75,70%	71,68%	6,22%
0.04	81	393	25	32	531	113	89,27%	76,42%	71,68%	5,98%
0.05	81	393	25	32	531	113	89,27%	76,42%	71,68%	5,98%
0.06	81	393	25	32	531	113	89,27%	76,42%	71,68%	5,98%
0.07	81	393	25	32	531	113	89,27%	76,42%	71,68%	5,98%
0.08	81	393	25	32	531	113	89,27%	76,42%	71,68%	5,98%
0.09	81	393	25	32	531	113	89,27%	76,42%	71,68%	5,98%
0.1	81	393	25	32	531	113	89,27%	76,42%	71,68%	5,98%
0.11	81	393	25	32	531	113	89,27%	76,42%	71,68%	5,98%
0.12	81	393	25	32	531	113	89,27%	76,42%	71,68%	5,98%
0.13	81	393	25	32	531	113	89,27%	76,42%	71,68%	5,98%
0.14	81	393	25	32	531	113	89,27%	76,42%	71,68%	5,98%
0.15	80	393	25	33	531	113	89,08%	76,19%	70,80%	5,98%
0.16	80	393	25	33	531	113	89,08%	76,19%	70,80%	5,98%
0.17	80	393	25	33	531	113	89,08%	76,19%	70,80%	5,98%
0.18	80	393	25	33	531	113	89,08%	76,19%	70,80%	5,98%
0.19	80	393	25	33	531	113	89,08%	76,19%	70,80%	5,98%
0.2	80	393	25	33	531	113	89,08%	76,19%	70,80%	5,98%
0.21	79	393	25	34	531	113	88,89%	75,96%	69,91%	5,98%
0.22	79	393	25	34	531	113	88,89%	75,96%	69,91%	5,98%
0.23	79	393	25	34	531	113	88,89%	75,96%	69,91%	5,98%
0.24	77	393	25	36	531	113	88,51%	75,49%	68,14%	5,98%
0.25	77	393	25	36	531	113	88,51%	75,49%	68,14%	5,98%
0.26	77	393	25	36	531	113	88,51%	75,49%	68,14%	5,98%
0.27	77	393	25	36	531	113	88,51%	75,49%	68,14%	5,98%
0.28	77	393	25	36	531	113	88,51%	75,49%	68,14%	5,98%
0.29	77	393	25	36	531	113	88,51%	75,49%	68,14%	5,98%
0.3	77	393	25	36	531	113	88,51%	75,49%	68,14%	5,98%
0.31	77	393	25	36	531	113	88,51%	75,49%	68,14%	5,98%
0.32	77	393	25	36	531	113	88,51%	75,49%	68,14%	5,98%
0.33	77	393	25	36	531	113	88,51%	75,49%	68,14%	5,98%
0.34	77	393	25	36	531	113	88,51%	75,49%	68,14%	5,98%
0.35	77	393	25	36	531	113	88,51%	75,49%	68,14%	5,98%
0.36	77	393	25	36	531	113	88,51%	75,49%	68,14%	5,98%
0.37	77	393	25	36	531	113	88,51%	75,49%	68,14%	5,98%
0.38	77	393	25	36	531	113	88,51%	75,49%	68,14%	5,98%
0.39	77	393	25	36	531	113	88,51%	75,49%	68,14%	5,98%
0.4	77	393	25	36	531	113	88,51%	75,49%	68,14%	5,98%
0.41	77	401	17	36	531	113	90,02%	81,91%	68,14%	4,07%
0.42	77	401	17	36	531	113	90,02%	81,91%	68,14%	4,07%
0.43	77	401	17	36	531	113	90,02%	81,91%	68,14%	4,07%
0.44	77	401	17	36	531	113	90,02%	81,91%	68,14%	4,07%
0.45	77	401	17	36	531	113	90,02%	81,91%	68,14%	4,07%
0.46	77	401	17	36	531	113	90,02%	81,91%	68,14%	4,07%
0.47	77	401	17	36	531	113	90,02%	81,91%	68,14%	4,07%
0.48	77	401	17	36	531	113	90,02%	81,91%	68,14%	4,07%
0.49	77	401	17	36	531	113	90,02%	81,91%	68,14%	4,07%
0.5	77	401	17	36	531	113	90,02%	81,91%	68,14%	4,07%
0.51	77	402	16	36	531	113	90,21%	82,80%	68,14%	3,83%
0.52	77	402	16	36	531	113	90,21%	82,80%	68,14%	3,83%
0.53	77	402	16	36	531	113	90,21%	82,80%	68,14%	3,83%
0.54	77	402	16	36	531	113	90,21%	82,80%	68,14%	3,83%
0.55	77	402	16	36	531	113	90,21%	82,80%	68,14%	3,83%
0.56	77	402	16	36	531	113	90,21%	82,80%	68,14%	3,83%
0.57	77	402	16	36	531	113	90,21%	82,80%	68,14%	3,83%
0.58	77	402	16	36	531	113	90,21%	82,80%	68,14%	3,83%
0.59	77	402	16	36	531	113	90,21%	82,80%	68,14%	3,83%

para	TP	TN	FP	FN	Total	Total Attachments	Accuracy	Precision	Recall	Fallout
0.6	77	402	16	36	531	113	90,21%	82,80%	68,14%	3,83%
0.61	77	402	16	36	531	113	90,21%	82,80%	68,14%	3,83%
0.62	77	402	16	36	531	113	90,21%	82,80%	68,14%	3,83%
0.63	77	406	12	36	531	113	90,96%	86,52%	68,14%	2,87%
0.64	77	406	12	36	531	113	90,96%	86,52%	68,14%	2,87%
0.65	77	408	10	36	531	113	91,34%	88,51%	68,14%	2,39%
0.66	77	408	10	36	531	113	91,34%	88,51%	68,14%	2,39%
0.67	77	408	10	36	531	113	91,34%	88,51%	68,14%	2,39%
0.68	77	408	10	36	531	113	91,34%	88,51%	68,14%	2,39%
0.69	77	408	10	36	531	113	91,34%	88,51%	68,14%	2,39%
0.7	77	408	10	36	531	113	91,34%	88,51%	68,14%	2,39%
0.71	77	408	10	36	531	113	91,34%	88,51%	68,14%	2,39%
0.72	77	408	10	36	531	113	91,34%	88,51%	68,14%	2,39%
0.73	77	408	10	36	531	113	91,34%	88,51%	68,14%	2,39%
0.74	76	408	10	37	531	113	91,15%	88,37%	67,26%	2,39%
0.75	76	408	10	37	531	113	91,15%	88,37%	67,26%	2,39%
0.76	76	408	10	37	531	113	91,15%	88,37%	67,26%	2,39%
0.77	75	408	10	38	531	113	90,96%	88,24%	66,37%	2,39%
0.78	75	408	10	38	531	113	90,96%	88,24%	66,37%	2,39%
0.79	75	408	10	38	531	113	90,96%	88,24%	66,37%	2,39%
0.8	75	408	10	38	531	113	90,96%	88,24%	66,37%	2,39%
0.81	69	408	10	44	531	113	89,83%	87,34%	61,06%	2,39%
0.82	64	408	10	49	531	113	88,89%	86,49%	56,64%	2,39%
0.83	62	408	10	51	531	113	88,51%	86,11%	54,87%	2,39%
0.84	62	408	10	51	531	113	88,51%	86,11%	54,87%	2,39%
0.85	62	408	10	51	531	113	88,51%	86,11%	54,87%	2,39%
0.86	61	408	10	52	531	113	88,32%	85,92%	53,98%	2,39%
0.87	59	408	10	54	531	113	87,95%	85,51%	52,21%	2,39%
0.88	59	408	10	54	531	113	87,95%	85,51%	52,21%	2,39%
0.89	59	408	10	54	531	113	87,95%	85,51%	52,21%	2,39%
0.9	59	408	10	54	531	113	87,95%	85,51%	52,21%	2,39%
0.91	59	408	10	54	531	113	87,95%	85,51%	52,21%	2,39%
0.92	59	408	10	54	531	113	87,95%	85,51%	52,21%	2,39%
0.93	59	408	10	54	531	113	87,95%	85,51%	52,21%	2,39%
0.94	59	408	10	54	531	113	87,95%	85,51%	52,21%	2,39%
0.95	59	408	10	54	531	113	87,95%	85,51%	52,21%	2,39%
0.96	59	408	10	54	531	113	87,95%	85,51%	52,21%	2,39%
0.97	59	408	10	54	531	113	87,95%	85,51%	52,21%	2,39%
0.98	59	409	9	54	531	113	88,14%	86,76%	52,21%	2,15%
0.99	59	409	9	54	531	113	88,14%	86,76%	52,21%	2,15%
1.00	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%

6.1.2.4 Naive Bayes

para	TP	TN	FP	FN	Total	Total Attachments	Accuracy	Precision	Recall	Fallout
0.0	113	0	418	0	531	113	21,28%	21,28%	100,00%	100,00%
0.01	113	0	418	0	531	113	21,28%	21,28%	100,00%	100,00%
0.02	113	0	418	0	531	113	21,28%	21,28%	100,00%	100,00%
0.03	113	0	418	0	531	113	21,28%	21,28%	100,00%	100,00%
0.04	113	0	418	0	531	113	21,28%	21,28%	100,00%	100,00%
0.05	113	0	418	0	531	113	21,28%	21,28%	100,00%	100,00%
0.06	113	0	418	0	531	113	21,28%	21,28%	100,00%	100,00%
0.07	113	1	417	0	531	113	21,47%	21,32%	100,00%	99,76%
0.08	113	1	417	0	531	113	21,47%	21,32%	100,00%	99,76%
0.09	113	1	417	0	531	113	21,47%	21,32%	100,00%	99,76%
0.1	113	1	417	0	531	113	21,47%	21,32%	100,00%	99,76%
0.11	113	1	417	0	531	113	21,47%	21,32%	100,00%	99,76%
0.12	113	1	417	0	531	113	21,47%	21,32%	100,00%	99,76%
0.13	113	1	417	0	531	113	21,47%	21,32%	100,00%	99,76%
0.14	113	1	417	0	531	113	21,47%	21,32%	100,00%	99,76%
0.15	113	1	417	0	531	113	21,47%	21,32%	100,00%	99,76%
0.16	113	1	417	0	531	113	21,47%	21,32%	100,00%	99,76%
0.17	113	1	417	0	531	113	21,47%	21,32%	100,00%	99,76%
0.18	113	1	417	0	531	113	21,47%	21,32%	100,00%	99,76%
0.19	113	1	417	0	531	113	21,47%	21,32%	100,00%	99,76%
0.2	113	1	417	0	531	113	21,47%	21,32%	100,00%	99,76%
0.21	113	1	417	0	531	113	21,47%	21,32%	100,00%	99,76%
0.22	113	1	417	0	531	113	21,47%	21,32%	100,00%	99,76%
0.23	113	1	417	0	531	113	21,47%	21,32%	100,00%	99,76%
0.24	113	1	417	0	531	113	21,47%	21,32%	100,00%	99,76%
0.25	113	1	417	0	531	113	21,47%	21,32%	100,00%	99,76%
0.26	113	1	417	0	531	113	21,47%	21,32%	100,00%	99,76%
0.27	113	1	417	0	531	113	21,47%	21,32%	100,00%	99,76%
0.28	113	1	417	0	531	113	21,47%	21,32%	100,00%	99,76%
0.29	113	1	417	0	531	113	21,47%	21,32%	100,00%	99,76%
0.3	113	1	417	0	531	113	21,47%	21,32%	100,00%	99,76%
0.31	113	1	417	0	531	113	21,47%	21,32%	100,00%	99,76%
0.32	113	1	417	0	531	113	21,47%	21,32%	100,00%	99,76%
0.33	113	1	417	0	531	113	21,47%	21,32%	100,00%	99,76%
0.34	113	1	417	0	531	113	21,47%	21,32%	100,00%	99,76%
0.35	113	1	417	0	531	113	21,47%	21,32%	100,00%	99,76%
0.36	113	1	417	0	531	113	21,47%	21,32%	100,00%	99,76%
0.37	113	1	417	0	531	113	21,47%	21,32%	100,00%	99,76%
0.38	113	1	417	0	531	113	21,47%	21,32%	100,00%	99,76%
0.39	113	1	417	0	531	113	21,47%	21,32%	100,00%	99,76%
0.4	113	1	417	0	531	113	21,47%	21,32%	100,00%	99,76%
0.41	113	1	417	0	531	113	21,47%	21,32%	100,00%	99,76%
0.42	113	1	417	0	531	113	21,47%	21,32%	100,00%	99,76%
0.43	113	2	416	0	531	113	21,66%	21,36%	100,00%	99,52%
0.44	113	3	415	0	531	113	21,85%	21,40%	100,00%	99,28%
0.45	113	3	415	0	531	113	21,85%	21,40%	100,00%	99,28%
0.46	112	24	394	1	531	113	25,61%	22,13%	99,12%	94,26%
0.47	107	137	281	6	531	113	45,95%	27,58%	94,69%	67,22%
0.48	99	279	139	14	531	113	71,19%	41,60%	87,61%	33,25%
0.49	95	364	54	18	531	113	86,44%	63,76%	84,07%	12,92%
0.5	87	389	29	26	531	113	89,64%	75,00%	76,99%	6,94%
0.51	70	407	11	43	531	113	89,83%	86,42%	61,95%	2,63%
0.52	58	414	4	55	531	113	88,89%	93,55%	51,33%	0,96%
0.53	54	417	1	59	531	113	88,70%	98,18%	47,79%	0,24%
0.54	49	417	1	64	531	113	87,76%	98,00%	43,36%	0,24%
0.55	42	417	1	71	531	113	86,44%	97,67%	37,17%	0,24%
0.56	30	417	1	83	531	113	84,18%	96,77%	26,55%	0,24%
0.57	23	417	1	90	531	113	82,86%	95,83%	20,35%	0,24%
0.58	21	417	1	92	531	113	82,49%	95,45%	18,58%	0,24%
0.59	16	418	0	97	531	113	81,73%	100,00%	14,16%	0,00%

para	TP	TN	FP	FN	Total	Total Attachments	Accuracy	Precision	Recall	Fallout
0.6	2	418	0	111	531	113	79,10%	100,00%	1,77%	0,00%
0.61	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.62	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.63	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.64	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.65	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.66	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.67	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.68	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.69	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.7	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.71	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.72	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.73	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.74	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.75	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.76	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.77	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.78	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.79	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.8	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.81	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.82	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.83	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.84	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.85	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.86	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.87	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.88	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.89	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.9	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.91	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.92	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.93	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.94	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.95	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.96	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.97	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.98	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.99	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
1.00	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%

6.1.2.5 Combined

para	TP	TN	FP	FN	Total	Total Attachments	Accuracy	Precision	Recall	Fallout
0.0	113	0	418	0	531	113	21,28%	21,28%	100,00%	100,00%
0.01	113	0	418	0	531	113	21,28%	21,28%	100,00%	100,00%
0.02	113	0	418	0	531	113	21,28%	21,28%	100,00%	100,00%
0.03	113	0	418	0	531	113	21,28%	21,28%	100,00%	100,00%
0.04	113	0	418	0	531	113	21,28%	21,28%	100,00%	100,00%
0.05	113	0	418	0	531	113	21,28%	21,28%	100,00%	100,00%
0.06	113	0	418	0	531	113	21,28%	21,28%	100,00%	100,00%
0.07	113	0	418	0	531	113	21,28%	21,28%	100,00%	100,00%
0.08	113	0	418	0	531	113	21,28%	21,28%	100,00%	100,00%
0.09	113	0	418	0	531	113	21,28%	21,28%	100,00%	100,00%
0.1	113	0	418	0	531	113	21,28%	21,28%	100,00%	100,00%
0.11	113	0	418	0	531	113	21,28%	21,28%	100,00%	100,00%
0.12	113	0	418	0	531	113	21,28%	21,28%	100,00%	100,00%
0.13	113	0	418	0	531	113	21,28%	21,28%	100,00%	100,00%
0.14	113	0	418	0	531	113	21,28%	21,28%	100,00%	100,00%
0.15	113	0	418	0	531	113	21,28%	21,28%	100,00%	100,00%
0.16	113	0	418	0	531	113	21,28%	21,28%	100,00%	100,00%
0.17	113	0	418	0	531	113	21,28%	21,28%	100,00%	100,00%
0.18	113	0	418	0	531	113	21,28%	21,28%	100,00%	100,00%
0.19	113	0	418	0	531	113	21,28%	21,28%	100,00%	100,00%
0.2	113	0	418	0	531	113	21,28%	21,28%	100,00%	100,00%
0.21	113	0	418	0	531	113	21,28%	21,28%	100,00%	100,00%
0.22	113	1	417	0	531	113	21,47%	21,32%	100,00%	99,76%
0.23	113	1	417	0	531	113	21,47%	21,32%	100,00%	99,76%
0.24	113	1	417	0	531	113	21,47%	21,32%	100,00%	99,76%
0.25	113	1	417	0	531	113	21,47%	21,32%	100,00%	99,76%
0.26	113	1	417	0	531	113	21,47%	21,32%	100,00%	99,76%
0.27	113	1	417	0	531	113	21,47%	21,32%	100,00%	99,76%
0.28	113	1	417	0	531	113	21,47%	21,32%	100,00%	99,76%
0.29	113	2	416	0	531	113	21,66%	21,36%	100,00%	99,52%
0.3	113	3	415	0	531	113	21,85%	21,40%	100,00%	99,28%
0.31	112	48	370	1	531	113	30,13%	23,24%	99,12%	88,52%
0.32	111	272	146	2	531	113	72,13%	43,19%	98,23%	34,93%
0.33	106	365	53	7	531	113	88,70%	66,67%	93,81%	12,68%
0.34	92	390	28	21	531	113	90,77%	76,67%	81,42%	6,70%
0.35	82	393	25	31	531	113	89,45%	76,64%	72,57%	5,98%
0.36	80	396	22	33	531	113	89,64%	78,43%	70,80%	5,26%
0.37	80	396	22	33	531	113	89,64%	78,43%	70,80%	5,26%
0.38	79	396	22	34	531	113	89,45%	78,22%	69,91%	5,26%
0.39	78	396	22	35	531	113	89,27%	78,00%	69,03%	5,26%
0.4	77	396	22	36	531	113	89,08%	77,78%	68,14%	5,26%
0.41	77	396	22	36	531	113	89,08%	77,78%	68,14%	5,26%
0.42	77	396	22	36	531	113	89,08%	77,78%	68,14%	5,26%
0.43	77	396	22	36	531	113	89,08%	77,78%	68,14%	5,26%
0.44	77	404	14	36	531	113	90,58%	84,62%	68,14%	3,35%
0.45	76	404	14	37	531	113	90,40%	84,44%	67,26%	3,35%
0.46	76	404	14	37	531	113	90,40%	84,44%	67,26%	3,35%
0.47	76	405	13	37	531	113	90,58%	85,39%	67,26%	3,11%
0.48	76	405	13	37	531	113	90,58%	85,39%	67,26%	3,11%
0.49	76	405	13	37	531	113	90,58%	85,39%	67,26%	3,11%
0.5	76	406	12	37	531	113	90,77%	86,36%	67,26%	2,87%
0.51	76	407	11	37	531	113	90,96%	87,36%	67,26%	2,63%
0.52	76	407	11	37	531	113	90,96%	87,36%	67,26%	2,63%
0.53	76	407	11	37	531	113	90,96%	87,36%	67,26%	2,63%
0.54	76	408	10	37	531	113	91,15%	88,37%	67,26%	2,39%
0.55	76	409	9	37	531	113	91,34%	89,41%	67,26%	2,15%
0.56	76	409	9	37	531	113	91,34%	89,41%	67,26%	2,15%
0.57	76	409	9	37	531	113	91,34%	89,41%	67,26%	2,15%
0.58	76	409	9	37	531	113	91,34%	89,41%	67,26%	2,15%
0.59	76	409	9	37	531	113	91,34%	89,41%	67,26%	2,15%

para	TP	TN	FP	FN	Total	Total Attachments	Accuracy	Precision	Recall	Fallout
0.6	76	410	8	37	531	113	91,53%	90,48%	67,26%	1,91%
0.61	68	410	8	45	531	113	90,02%	89,47%	60,18%	1,91%
0.62	63	410	8	50	531	113	89,08%	88,73%	55,75%	1,91%
0.63	62	410	8	51	531	113	88,89%	88,57%	54,87%	1,91%
0.64	61	410	8	52	531	113	88,70%	88,41%	53,98%	1,91%
0.65	61	410	8	52	531	113	88,70%	88,41%	53,98%	1,91%
0.66	60	411	7	53	531	113	88,70%	89,55%	53,10%	1,67%
0.67	58	414	4	55	531	113	88,89%	93,55%	51,33%	0,96%
0.68	57	417	1	56	531	113	89,27%	98,28%	50,44%	0,24%
0.69	52	417	1	61	531	113	88,32%	98,11%	46,02%	0,24%
0.7	45	417	1	68	531	113	87,01%	97,83%	39,82%	0,24%
0.71	28	417	1	85	531	113	83,80%	96,55%	24,78%	0,24%
0.72	23	417	1	90	531	113	82,86%	95,83%	20,35%	0,24%
0.73	16	418	0	97	531	113	81,73%	100,00%	14,16%	0,00%
0.74	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.75	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.76	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.77	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.78	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.79	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.8	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.81	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.82	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.83	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.84	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.85	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.86	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.87	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.88	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.89	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.9	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.91	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.92	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.93	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.94	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.95	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.96	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.97	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.98	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
0.99	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%
1.00	0	418	0	113	531	113	78,72%	100,00%	0,00%	0,00%

6.1.3 Datensatz 3

6.1.3.1 Einfacher Algorithmus

para	TP	TN	FP	FN	Total	Total Attachments	Accuracy	Precision	Recall	Fallout
1	202	1392	121	530	2.245	732	71,00%	62,54%	27,60%	18,59%

6.1.3.2 Erweiterter Algorithmus

para	TP	TN	FP	FN	Total	Total Attachments	Accuracy	Precision	Recall	Fallout
2	732	952	561	0	2.245	732	75,01%	56,61%	100,00%	37,08%
3	368	1193	320	364	2.245	732	69,53%	53,49%	50,27%	21,15%
4	342	1297	216	390	2.245	732	73,01%	61,29%	46,72%	14,28%
5	303	1356	157	429	2.245	732	73,90%	65,87%	41,39%	10,38%
6	280	1390	123	452	2.245	732	74,39%	69,48%	38,25%	8,13%
7	259	1409	104	473	2.245	732	74,30%	71,35%	35,38%	6,87%
8	243	1424	89	489	2.245	732	74,25%	73,19%	33,20%	5,88%
9	240	1428	85	492	2.245	732	74,30%	73,85%	32,79%	5,62%
10	234	1433	80	498	2.245	732	74,25%	74,52%	31,97%	5,29%
11	229	1443	70	503	2.245	732	74,48%	76,59%	31,28%	4,63%
12	227	1446	67	505	2.245	732	74,52%	77,21%	31,01%	4,43%
13	223	1449	64	509	2.245	732	74,48%	77,70%	30,46%	4,23%
14	222	1451	62	510	2.245	732	74,52%	78,17%	30,33%	4,10%
15	222	1454	59	510	2.245	732	74,65%	79,00%	30,33%	3,90%

6.1.3.3 Paul Graham

para	TP	TN	FP	FN	Total	Total Attachments	Accuracy	Precision	Recall	Fallout
0.0	732	0	1513	0	2.245	732	32,61%	32,61%	100,00%	100,00%
0.01	500	1410	103	232	2.245	732	85,08%	82,92%	68,31%	6,81%
0.02	491	1415	98	241	2.245	732	84,90%	83,36%	67,08%	6,48%
0.03	484	1416	97	248	2.245	732	84,63%	83,30%	66,12%	6,41%
0.04	477	1416	97	255	2.245	732	84,32%	83,10%	65,16%	6,41%
0.05	474	1421	92	258	2.245	732	84,41%	83,75%	64,75%	6,08%
0.06	470	1422	91	262	2.245	732	84,28%	83,78%	64,21%	6,01%
0.07	469	1422	91	263	2.245	732	84,23%	83,75%	64,07%	6,01%
0.08	468	1422	91	264	2.245	732	84,19%	83,72%	63,93%	6,01%
0.09	468	1424	89	264	2.245	732	84,28%	84,02%	63,93%	5,88%
0.1	466	1425	88	266	2.245	732	84,23%	84,12%	63,66%	5,82%
0.11	466	1426	87	266	2.245	732	84,28%	84,27%	63,66%	5,75%
0.12	466	1427	86	266	2.245	732	84,32%	84,42%	63,66%	5,68%
0.13	465	1427	86	267	2.245	732	84,28%	84,39%	63,52%	5,68%
0.14	461	1427	86	271	2.245	732	84,10%	84,28%	62,98%	5,68%
0.15	461	1427	86	271	2.245	732	84,10%	84,28%	62,98%	5,68%
0.16	461	1427	86	271	2.245	732	84,10%	84,28%	62,98%	5,68%
0.17	458	1428	85	274	2.245	732	84,01%	84,35%	62,57%	5,62%
0.18	458	1429	84	274	2.245	732	84,05%	84,50%	62,57%	5,55%
0.19	457	1429	84	275	2.245	732	84,01%	84,47%	62,43%	5,55%
0.2	455	1430	83	277	2.245	732	83,96%	84,57%	62,16%	5,49%
0.21	454	1430	83	278	2.245	732	83,92%	84,54%	62,02%	5,49%
0.22	454	1431	82	278	2.245	732	83,96%	84,70%	62,02%	5,42%
0.23	453	1431	82	279	2.245	732	83,92%	84,67%	61,89%	5,42%
0.24	453	1431	82	279	2.245	732	83,92%	84,67%	61,89%	5,42%
0.25	453	1431	82	279	2.245	732	83,92%	84,67%	61,89%	5,42%
0.26	453	1431	82	279	2.245	732	83,92%	84,67%	61,89%	5,42%
0.27	451	1432	81	281	2.245	732	83,88%	84,77%	61,61%	5,35%
0.28	449	1433	80	283	2.245	732	83,83%	84,88%	61,34%	5,29%
0.29	448	1433	80	284	2.245	732	83,79%	84,85%	61,20%	5,29%
0.3	448	1433	80	284	2.245	732	83,79%	84,85%	61,20%	5,29%
0.31	448	1433	80	284	2.245	732	83,79%	84,85%	61,20%	5,29%
0.32	447	1433	80	285	2.245	732	83,74%	84,82%	61,07%	5,29%
0.33	447	1434	79	285	2.245	732	83,79%	84,98%	61,07%	5,22%
0.34	447	1434	79	285	2.245	732	83,79%	84,98%	61,07%	5,22%
0.35	447	1435	78	285	2.245	732	83,83%	85,14%	61,07%	5,16%
0.36	447	1435	78	285	2.245	732	83,83%	85,14%	61,07%	5,16%
0.37	447	1440	73	285	2.245	732	84,05%	85,96%	61,07%	4,82%
0.38	446	1441	72	286	2.245	732	84,05%	86,10%	60,93%	4,76%
0.39	445	1446	67	287	2.245	732	84,23%	86,91%	60,79%	4,43%
0.4	443	1446	67	289	2.245	732	84,14%	86,86%	60,52%	4,43%
0.41	443	1463	50	289	2.245	732	84,90%	89,86%	60,52%	3,30%
0.42	442	1463	50	290	2.245	732	84,86%	89,84%	60,38%	3,30%
0.43	441	1463	50	291	2.245	732	84,81%	89,82%	60,25%	3,30%
0.44	441	1463	50	291	2.245	732	84,81%	89,82%	60,25%	3,30%
0.45	441	1463	50	291	2.245	732	84,81%	89,82%	60,25%	3,30%
0.46	440	1463	50	292	2.245	732	84,77%	89,80%	60,11%	3,30%
0.47	439	1463	50	293	2.245	732	84,72%	89,78%	59,97%	3,30%
0.48	438	1463	50	294	2.245	732	84,68%	89,75%	59,84%	3,30%
0.49	438	1463	50	294	2.245	732	84,68%	89,75%	59,84%	3,30%
0.5	437	1463	50	295	2.245	732	84,63%	89,73%	59,70%	3,30%
0.51	437	1490	23	295	2.245	732	85,84%	95,00%	59,70%	1,52%
0.52	437	1490	23	295	2.245	732	85,84%	95,00%	59,70%	1,52%
0.53	436	1490	23	296	2.245	732	85,79%	94,99%	59,56%	1,52%
0.54	436	1490	23	296	2.245	732	85,79%	94,99%	59,56%	1,52%
0.55	433	1490	23	299	2.245	732	85,66%	94,96%	59,15%	1,52%
0.56	432	1490	23	300	2.245	732	85,61%	94,95%	59,02%	1,52%
0.57	431	1490	23	301	2.245	732	85,57%	94,93%	58,88%	1,52%
0.58	431	1490	23	301	2.245	732	85,57%	94,93%	58,88%	1,52%
0.59	430	1490	23	302	2.245	732	85,52%	94,92%	58,74%	1,52%

para	TP	TN	FP	FN	Total	Total Attachments	Accuracy	Precision	Recall	Fallout
0.6	430	1490	23	302	2.245	732	85,52%	94,92%	58,74%	1,52%
0.61	427	1490	23	305	2.245	732	85,39%	94,89%	58,33%	1,52%
0.62	426	1490	23	306	2.245	732	85,35%	94,88%	58,20%	1,52%
0.63	425	1492	21	307	2.245	732	85,39%	95,29%	58,06%	1,39%
0.64	425	1492	21	307	2.245	732	85,39%	95,29%	58,06%	1,39%
0.65	422	1492	21	310	2.245	732	85,26%	95,26%	57,65%	1,39%
0.66	422	1492	21	310	2.245	732	85,26%	95,26%	57,65%	1,39%
0.67	422	1492	21	310	2.245	732	85,26%	95,26%	57,65%	1,39%
0.68	422	1492	21	310	2.245	732	85,26%	95,26%	57,65%	1,39%
0.69	422	1492	21	310	2.245	732	85,26%	95,26%	57,65%	1,39%
0.7	422	1492	21	310	2.245	732	85,26%	95,26%	57,65%	1,39%
0.71	422	1492	21	310	2.245	732	85,26%	95,26%	57,65%	1,39%
0.72	420	1492	21	312	2.245	732	85,17%	95,24%	57,38%	1,39%
0.73	418	1492	21	314	2.245	732	85,08%	95,22%	57,10%	1,39%
0.74	418	1492	21	314	2.245	732	85,08%	95,22%	57,10%	1,39%
0.75	415	1492	21	317	2.245	732	84,94%	95,18%	56,69%	1,39%
0.76	412	1492	21	320	2.245	732	84,81%	95,15%	56,28%	1,39%
0.77	412	1492	21	320	2.245	732	84,81%	95,15%	56,28%	1,39%
0.78	411	1492	21	321	2.245	732	84,77%	95,14%	56,15%	1,39%
0.79	408	1492	21	324	2.245	732	84,63%	95,10%	55,74%	1,39%
0.8	406	1492	21	326	2.245	732	84,54%	95,08%	55,46%	1,39%
0.81	403	1492	21	329	2.245	732	84,41%	95,05%	55,05%	1,39%
0.82	403	1493	20	329	2.245	732	84,45%	95,27%	55,05%	1,32%
0.83	384	1493	20	348	2.245	732	83,61%	95,05%	52,46%	1,32%
0.84	342	1494	19	390	2.245	732	81,78%	94,74%	46,72%	1,26%
0.85	316	1494	19	416	2.245	732	80,62%	94,33%	43,17%	1,26%
0.86	315	1494	19	417	2.245	732	80,58%	94,31%	43,03%	1,26%
0.87	298	1494	19	434	2.245	732	79,82%	94,01%	40,71%	1,26%
0.88	296	1494	19	436	2.245	732	79,73%	93,97%	40,44%	1,26%
0.89	295	1494	19	437	2.245	732	79,69%	93,95%	40,30%	1,26%
0.9	295	1494	19	437	2.245	732	79,69%	93,95%	40,30%	1,26%
0.91	292	1494	19	440	2.245	732	79,55%	93,89%	39,89%	1,26%
0.92	288	1494	19	444	2.245	732	79,38%	93,81%	39,34%	1,26%
0.93	286	1494	19	446	2.245	732	79,29%	93,77%	39,07%	1,26%
0.94	280	1494	19	452	2.245	732	79,02%	93,65%	38,25%	1,26%
0.95	278	1494	19	454	2.245	732	78,93%	93,60%	37,98%	1,26%
0.96	274	1494	19	458	2.245	732	78,75%	93,52%	37,43%	1,26%
0.97	270	1494	19	462	2.245	732	78,57%	93,43%	36,89%	1,26%
0.98	264	1494	19	468	2.245	732	78,31%	93,29%	36,07%	1,26%
0.99	254	1494	19	478	2.245	732	77,86%	93,04%	34,70%	1,26%
1.00	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%

6.1.3.4 Naive Bayes

para	TP	TN	FP	FN	Total	Total Attachments	Accuracy	Precision	Recall	Fallout
0.0	732	0	1513	0	2.245	732	32,61%	32,61%	100,00%	100,00%
0.01	732	0	1513	0	2.245	732	32,61%	32,61%	100,00%	100,00%
0.02	732	0	1513	0	2.245	732	32,61%	32,61%	100,00%	100,00%
0.03	732	0	1513	0	2.245	732	32,61%	32,61%	100,00%	100,00%
0.04	732	0	1513	0	2.245	732	32,61%	32,61%	100,00%	100,00%
0.05	732	0	1513	0	2.245	732	32,61%	32,61%	100,00%	100,00%
0.06	732	0	1513	0	2.245	732	32,61%	32,61%	100,00%	100,00%
0.07	732	0	1513	0	2.245	732	32,61%	32,61%	100,00%	100,00%
0.08	732	0	1513	0	2.245	732	32,61%	32,61%	100,00%	100,00%
0.09	732	0	1513	0	2.245	732	32,61%	32,61%	100,00%	100,00%
0.1	732	0	1513	0	2.245	732	32,61%	32,61%	100,00%	100,00%
0.11	732	17	1496	0	2.245	732	33,36%	32,85%	100,00%	100,00%
0.12	732	27	1486	0	2.245	732	33,81%	33,00%	100,00%	100,00%
0.13	732	27	1486	0	2.245	732	33,81%	33,00%	100,00%	100,00%
0.14	732	27	1486	0	2.245	732	33,81%	33,00%	100,00%	100,00%
0.15	732	27	1486	0	2.245	732	33,81%	33,00%	100,00%	100,00%
0.16	732	27	1486	0	2.245	732	33,81%	33,00%	100,00%	100,00%
0.17	732	27	1486	0	2.245	732	33,81%	33,00%	100,00%	100,00%
0.18	732	27	1486	0	2.245	732	33,81%	33,00%	100,00%	100,00%
0.19	732	27	1486	0	2.245	732	33,81%	33,00%	100,00%	100,00%
0.2	732	27	1486	0	2.245	732	33,81%	33,00%	100,00%	100,00%
0.21	732	27	1486	0	2.245	732	33,81%	33,00%	100,00%	100,00%
0.22	732	27	1486	0	2.245	732	33,81%	33,00%	100,00%	100,00%
0.23	732	27	1486	0	2.245	732	33,81%	33,00%	100,00%	100,00%
0.24	732	27	1486	0	2.245	732	33,81%	33,00%	100,00%	100,00%
0.25	732	27	1486	0	2.245	732	33,81%	33,00%	100,00%	100,00%
0.26	732	27	1486	0	2.245	732	33,81%	33,00%	100,00%	100,00%
0.27	732	27	1486	0	2.245	732	33,81%	33,00%	100,00%	100,00%
0.28	732	27	1486	0	2.245	732	33,81%	33,00%	100,00%	100,00%
0.29	732	27	1486	0	2.245	732	33,81%	33,00%	100,00%	100,00%
0.3	732	27	1486	0	2.245	732	33,81%	33,00%	100,00%	100,00%
0.31	732	27	1486	0	2.245	732	33,81%	33,00%	100,00%	100,00%
0.32	732	27	1486	0	2.245	732	33,81%	33,00%	100,00%	100,00%
0.33	732	27	1486	0	2.245	732	33,81%	33,00%	100,00%	100,00%
0.34	732	27	1486	0	2.245	732	33,81%	33,00%	100,00%	100,00%
0.35	732	27	1486	0	2.245	732	33,81%	33,00%	100,00%	100,00%
0.36	732	27	1486	0	2.245	732	33,81%	33,00%	100,00%	100,00%
0.37	732	27	1486	0	2.245	732	33,81%	33,00%	100,00%	100,00%
0.38	732	27	1486	0	2.245	732	33,81%	33,00%	100,00%	100,00%
0.39	732	27	1486	0	2.245	732	33,81%	33,00%	100,00%	100,00%
0.4	732	27	1486	0	2.245	732	33,81%	33,00%	100,00%	100,00%
0.41	732	27	1486	0	2.245	732	33,81%	33,00%	100,00%	100,00%
0.42	732	27	1486	0	2.245	732	33,81%	33,00%	100,00%	100,00%
0.43	732	28	1485	0	2.245	732	33,85%	33,02%	100,00%	100,00%
0.44	732	28	1485	0	2.245	732	33,85%	33,02%	100,00%	100,00%
0.45	732	30	1483	0	2.245	732	33,94%	33,05%	100,00%	100,00%
0.46	732	34	1479	0	2.245	732	34,12%	33,11%	100,00%	100,00%
0.47	732	55	1458	0	2.245	732	35,06%	33,42%	100,00%	100,00%
0.48	732	214	1299	0	2.245	732	42,14%	36,04%	100,00%	100,00%
0.49	710	738	775	22	2.245	732	64,50%	47,81%	96,99%	97,24%
0.5	585	1412	101	147	2.245	732	88,95%	85,28%	79,92%	40,73%
0.51	429	1494	19	303	2.245	732	85,66%	95,76%	58,61%	5,90%
0.52	316	1506	7	416	2.245	732	81,16%	97,83%	43,17%	1,65%
0.53	258	1507	6	474	2.245	732	78,62%	97,73%	35,25%	1,25%
0.54	137	1509	4	595	2.245	732	73,32%	97,16%	18,72%	0,67%
0.55	105	1509	4	627	2.245	732	71,89%	96,33%	14,34%	0,63%
0.56	88	1509	4	644	2.245	732	71,14%	95,65%	12,02%	0,62%
0.57	63	1511	2	669	2.245	732	70,11%	96,92%	8,61%	0,30%
0.58	8	1513	0	724	2.245	732	67,75%	100,00%	1,09%	0,00%
0.59	1	1513	0	731	2.245	732	67,44%	100,00%	0,14%	0,00%

para	TP	TN	FP	FN	Total	Total Attachments	Accuracy	Precision	Recall	Fallout
0.6	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.61	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.62	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.63	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.64	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.65	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.66	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.67	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.68	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.69	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.7	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.71	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.72	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.73	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.74	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.75	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.76	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.77	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.78	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.79	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.8	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.81	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.82	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.83	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.84	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.85	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.86	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.87	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.88	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.89	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.9	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.91	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.92	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.93	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.94	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.95	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.96	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.97	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.98	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.99	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
1.00	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%

6.1.3.5 Combined

para	TP	TN	FP	FN	Total	Total Attachments	Accuracy	Precision	Recall	Fallout
0.0	732	0	1513	0	2.245	732	32,61%	32,61%	100,00%	100,00%
0.01	732	0	1513	0	2.245	732	32,61%	32,61%	100,00%	100,00%
0.02	732	0	1513	0	2.245	732	32,61%	32,61%	100,00%	100,00%
0.03	732	0	1513	0	2.245	732	32,61%	32,61%	100,00%	100,00%
0.04	732	0	1513	0	2.245	732	32,61%	32,61%	100,00%	100,00%
0.05	732	0	1513	0	2.245	732	32,61%	32,61%	100,00%	100,00%
0.06	732	0	1513	0	2.245	732	32,61%	32,61%	100,00%	100,00%
0.07	732	0	1513	0	2.245	732	32,61%	32,61%	100,00%	100,00%
0.08	732	0	1513	0	2.245	732	32,61%	32,61%	100,00%	100,00%
0.09	732	0	1513	0	2.245	732	32,61%	32,61%	100,00%	100,00%
0.1	732	0	1513	0	2.245	732	32,61%	32,61%	100,00%	100,00%
0.11	732	0	1513	0	2.245	732	32,61%	32,61%	100,00%	100,00%
0.12	732	0	1513	0	2.245	732	32,61%	32,61%	100,00%	100,00%
0.13	732	0	1513	0	2.245	732	32,61%	32,61%	100,00%	100,00%
0.14	732	0	1513	0	2.245	732	32,61%	32,61%	100,00%	100,00%
0.15	732	0	1513	0	2.245	732	32,61%	32,61%	100,00%	100,00%
0.16	732	0	1513	0	2.245	732	32,61%	32,61%	100,00%	100,00%
0.17	732	0	1513	0	2.245	732	32,61%	32,61%	100,00%	100,00%
0.18	732	0	1513	0	2.245	732	32,61%	32,61%	100,00%	100,00%
0.19	732	0	1513	0	2.245	732	32,61%	32,61%	100,00%	100,00%
0.2	732	0	1513	0	2.245	732	32,61%	32,61%	100,00%	100,00%
0.21	732	0	1513	0	2.245	732	32,61%	32,61%	100,00%	100,00%
0.22	732	0	1513	0	2.245	732	32,61%	32,61%	100,00%	100,00%
0.23	732	0	1513	0	2.245	732	32,61%	32,61%	100,00%	100,00%
0.24	732	16	1497	0	2.245	732	33,32%	32,84%	100,00%	98,94%
0.25	732	27	1486	0	2.245	732	33,81%	33,00%	100,00%	98,22%
0.26	732	27	1486	0	2.245	732	33,81%	33,00%	100,00%	98,22%
0.27	732	27	1486	0	2.245	732	33,81%	33,00%	100,00%	98,22%
0.28	732	28	1485	0	2.245	732	33,85%	33,02%	100,00%	98,15%
0.29	732	28	1485	0	2.245	732	33,85%	33,02%	100,00%	98,15%
0.3	732	28	1485	0	2.245	732	33,85%	33,02%	100,00%	98,15%
0.31	732	33	1480	0	2.245	732	34,08%	33,09%	100,00%	97,82%
0.32	732	183	1330	0	2.245	732	40,76%	35,50%	100,00%	87,90%
0.33	720	1183	330	12	2.245	732	84,77%	68,57%	98,36%	21,81%
0.34	550	1438	75	182	2.245	732	88,55%	88,00%	75,14%	4,96%
0.35	497	1448	65	235	2.245	732	86,64%	88,43%	67,90%	4,30%
0.36	479	1452	61	253	2.245	732	86,01%	88,70%	65,44%	4,03%
0.37	470	1453	60	262	2.245	732	85,66%	88,68%	64,21%	3,97%
0.38	467	1458	55	265	2.245	732	85,75%	89,46%	63,80%	3,64%
0.39	463	1459	54	269	2.245	732	85,61%	89,56%	63,25%	3,57%
0.4	460	1459	54	272	2.245	732	85,48%	89,49%	62,84%	3,57%
0.41	453	1459	54	279	2.245	732	85,17%	89,35%	61,89%	3,57%
0.42	451	1461	52	281	2.245	732	85,17%	89,66%	61,61%	3,44%
0.43	451	1465	48	281	2.245	732	85,35%	90,38%	61,61%	3,17%
0.44	448	1473	40	284	2.245	732	85,57%	91,80%	61,20%	2,64%
0.45	448	1491	22	284	2.245	732	86,37%	95,32%	61,20%	1,45%
0.46	445	1491	22	287	2.245	732	86,24%	95,29%	60,79%	1,45%
0.47	443	1491	22	289	2.245	732	86,15%	95,27%	60,52%	1,45%
0.48	441	1491	22	291	2.245	732	86,06%	95,25%	60,25%	1,45%
0.49	439	1491	22	293	2.245	732	85,97%	95,23%	59,97%	1,45%
0.5	437	1491	22	295	2.245	732	85,88%	95,21%	59,70%	1,45%
0.51	436	1491	22	296	2.245	732	85,84%	95,20%	59,56%	1,45%
0.52	436	1491	22	296	2.245	732	85,84%	95,20%	59,56%	1,45%
0.53	436	1491	22	296	2.245	732	85,84%	95,20%	59,56%	1,45%
0.54	433	1491	22	299	2.245	732	85,70%	95,16%	59,15%	1,45%
0.55	433	1491	22	299	2.245	732	85,70%	95,16%	59,15%	1,45%
0.56	431	1491	22	301	2.245	732	85,61%	95,14%	58,88%	1,45%
0.57	428	1491	22	304	2.245	732	85,48%	95,11%	58,47%	1,45%
0.58	422	1493	20	310	2.245	732	85,30%	95,48%	57,65%	1,32%
0.59	418	1493	20	314	2.245	732	85,12%	95,43%	57,10%	1,32%

para	TP	TN	FP	FN	Total	Total Attachments	Accuracy	Precision	Recall	Fallout
0.6	415	1493	20	317	2.245	732	84,99%	95,40%	56,69%	1,32%
0.61	413	1493	20	319	2.245	732	84,90%	95,38%	56,42%	1,32%
0.62	408	1494	19	324	2.245	732	84,72%	95,55%	55,74%	1,26%
0.63	403	1494	19	329	2.245	732	84,50%	95,50%	55,05%	1,26%
0.64	326	1495	18	406	2.245	732	81,11%	94,77%	44,54%	1,19%
0.65	298	1495	18	434	2.245	732	79,87%	94,30%	40,71%	1,19%
0.66	291	1495	18	441	2.245	732	79,55%	94,17%	39,75%	1,19%
0.67	271	1500	13	461	2.245	732	78,89%	95,42%	37,02%	0,86%
0.68	191	1508	5	541	2.245	732	75,68%	97,45%	26,09%	0,33%
0.69	149	1509	4	583	2.245	732	73,85%	97,39%	20,36%	0,26%
0.7	109	1509	4	623	2.245	732	72,07%	96,46%	14,89%	0,26%
0.71	94	1510	3	638	2.245	732	71,45%	96,91%	12,84%	0,20%
0.72	11	1513	0	721	2.245	732	67,88%	100,00%	1,50%	0,00%
0.73	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.74	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.75	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.76	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.77	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.78	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.79	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.8	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.81	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.82	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.83	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.84	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.85	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.86	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.87	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.88	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.89	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.9	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.91	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.92	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.93	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.94	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.95	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.96	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.97	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.98	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
0.99	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%
1.00	0	1513	0	732	2.245	732	67,39%	100,00%	0,00%	0,00%

6.1.4 Datensatz 4

6.1.4.1 Einfacher Algorithmus

para	TP	TN	FP	FN	Total	Total Attachments	Accuracy	Precision	Recall	Fallout
1	14	172	40	56	282	70	65,96%	25,93%	20,00%	18,87%
hline										

6.1.4.2 Erweiterter Algorithmus

para	TP	TN	FP	FN	Total	Total Attachments	Accuracy	Precision	Recall	Fallout
2	26	168	44	44	282	70	68,79%	37,14%	37,14%	20,75%
3	21	187	25	49	282	70	73,76%	45,65%	30,00%	11,79%
4	21	198	14	49	282	70	77,66%	60,00%	30,00%	6,60%
5	22	199	13	48	282	70	78,37%	62,86%	31,43%	6,13%
6	19	200	12	51	282	70	77,66%	61,29%	27,14%	5,66%
7	18	204	8	52	282	70	78,72%	69,23%	25,71%	3,77%
8	12	207	5	58	282	70	77,66%	70,59%	17,14%	2,36%
9	12	206	6	58	282	70	77,30%	66,67%	17,14%	2,83%
10	11	209	3	59	282	70	78,01%	78,57%	15,71%	1,42%
11	11	209	3	59	282	70	78,01%	78,57%	15,71%	1,42%
12	11	209	3	59	282	70	78,01%	78,57%	15,71%	1,42%
13	11	209	3	59	282	70	78,01%	78,57%	15,71%	1,42%
14	11	209	3	59	282	70	78,01%	78,57%	15,71%	1,42%
15	11	209	3	59	282	70	78,01%	78,57%	15,71%	1,42%

6.1.4.3 Paul Graham

para	TP	TN	FP	FN	Total	Total Attachments	Accuracy	Precision	Recall	Fallout
0.0	70	0	212	0	282	70	24,82%	24,82%	100,00%	100,00%
0.01	35	197	15	35	282	70	82,27%	70,00%	50,00%	7,08%
0.02	35	198	14	35	282	70	82,62%	71,43%	50,00%	6,60%
0.03	34	198	14	36	282	70	82,27%	70,83%	48,57%	6,60%
0.04	34	198	14	36	282	70	82,27%	70,83%	48,57%	6,60%
0.05	34	199	13	36	282	70	82,62%	72,34%	48,57%	6,13%
0.06	34	200	12	36	282	70	82,98%	73,91%	48,57%	5,66%
0.07	34	200	12	36	282	70	82,98%	73,91%	48,57%	5,66%
0.08	34	200	12	36	282	70	82,98%	73,91%	48,57%	5,66%
0.09	34	200	12	36	282	70	82,98%	73,91%	48,57%	5,66%
0.1	34	200	12	36	282	70	82,98%	73,91%	48,57%	5,66%
0.11	34	201	11	36	282	70	83,33%	75,56%	48,57%	5,19%
0.12	34	201	11	36	282	70	83,33%	75,56%	48,57%	5,19%
0.13	34	201	11	36	282	70	83,33%	75,56%	48,57%	5,19%
0.14	34	201	11	36	282	70	83,33%	75,56%	48,57%	5,19%
0.15	34	201	11	36	282	70	83,33%	75,56%	48,57%	5,19%
0.16	34	201	11	36	282	70	83,33%	75,56%	48,57%	5,19%
0.17	34	201	11	36	282	70	83,33%	75,56%	48,57%	5,19%
0.18	34	201	11	36	282	70	83,33%	75,56%	48,57%	5,19%
0.19	34	201	11	36	282	70	83,33%	75,56%	48,57%	5,19%
0.2	34	201	11	36	282	70	83,33%	75,56%	48,57%	5,19%
0.21	34	201	11	36	282	70	83,33%	75,56%	48,57%	5,19%
0.22	34	201	11	36	282	70	83,33%	75,56%	48,57%	5,19%
0.23	34	201	11	36	282	70	83,33%	75,56%	48,57%	5,19%
0.24	34	201	11	36	282	70	83,33%	75,56%	48,57%	5,19%
0.25	34	201	11	36	282	70	83,33%	75,56%	48,57%	5,19%
0.26	34	201	11	36	282	70	83,33%	75,56%	48,57%	5,19%
0.27	34	201	11	36	282	70	83,33%	75,56%	48,57%	5,19%
0.28	34	201	11	36	282	70	83,33%	75,56%	48,57%	5,19%
0.29	34	201	11	36	282	70	83,33%	75,56%	48,57%	5,19%
0.3	34	201	11	36	282	70	83,33%	75,56%	48,57%	5,19%
0.31	34	201	11	36	282	70	83,33%	75,56%	48,57%	5,19%
0.32	34	201	11	36	282	70	83,33%	75,56%	48,57%	5,19%
0.33	34	201	11	36	282	70	83,33%	75,56%	48,57%	5,19%
0.34	34	201	11	36	282	70	83,33%	75,56%	48,57%	5,19%
0.35	34	201	11	36	282	70	83,33%	75,56%	48,57%	5,19%
0.36	34	201	11	36	282	70	83,33%	75,56%	48,57%	5,19%
0.37	34	201	11	36	282	70	83,33%	75,56%	48,57%	5,19%
0.38	34	201	11	36	282	70	83,33%	75,56%	48,57%	5,19%
0.39	34	201	11	36	282	70	83,33%	75,56%	48,57%	5,19%
0.4	34	201	11	36	282	70	83,33%	75,56%	48,57%	5,19%
0.41	34	202	10	36	282	70	83,69%	77,27%	48,57%	4,72%
0.42	34	202	10	36	282	70	83,69%	77,27%	48,57%	4,72%
0.43	34	202	10	36	282	70	83,69%	77,27%	48,57%	4,72%
0.44	34	202	10	36	282	70	83,69%	77,27%	48,57%	4,72%
0.45	34	202	10	36	282	70	83,69%	77,27%	48,57%	4,72%
0.46	34	202	10	36	282	70	83,69%	77,27%	48,57%	4,72%
0.47	34	202	10	36	282	70	83,69%	77,27%	48,57%	4,72%
0.48	34	202	10	36	282	70	83,69%	77,27%	48,57%	4,72%
0.49	34	202	10	36	282	70	83,69%	77,27%	48,57%	4,72%
0.5	34	202	10	36	282	70	83,69%	77,27%	48,57%	4,72%
0.51	33	205	7	37	282	70	84,40%	82,50%	47,14%	3,30%
0.52	33	205	7	37	282	70	84,40%	82,50%	47,14%	3,30%
0.53	31	205	7	39	282	70	83,69%	81,58%	44,29%	3,30%
0.54	24	205	7	46	282	70	81,21%	77,42%	34,29%	3,30%
0.55	18	207	5	52	282	70	79,79%	78,26%	25,71%	2,36%
0.56	16	207	5	54	282	70	79,08%	76,19%	22,86%	2,36%
0.57	13	207	5	57	282	70	78,01%	72,22%	18,57%	2,36%
0.58	13	207	5	57	282	70	78,01%	72,22%	18,57%	2,36%
0.59	13	207	5	57	282	70	78,01%	72,22%	18,57%	2,36%

para	TP	TN	FP	FN	Total	Total Attachments	Accuracy	Precision	Recall	Fallout
0.6	13	207	5	57	282	70	78,01%	72,22%	18,57%	2,36%
0.61	13	208	4	57	282	70	78,37%	76,47%	18,57%	1,89%
0.62	13	208	4	57	282	70	78,37%	76,47%	18,57%	1,89%
0.63	13	208	4	57	282	70	78,37%	76,47%	18,57%	1,89%
0.64	13	208	4	57	282	70	78,37%	76,47%	18,57%	1,89%
0.65	13	208	4	57	282	70	78,37%	76,47%	18,57%	1,89%
0.66	13	208	4	57	282	70	78,37%	76,47%	18,57%	1,89%
0.67	13	208	4	57	282	70	78,37%	76,47%	18,57%	1,89%
0.68	13	208	4	57	282	70	78,37%	76,47%	18,57%	1,89%
0.69	13	208	4	57	282	70	78,37%	76,47%	18,57%	1,89%
0.7	13	208	4	57	282	70	78,37%	76,47%	18,57%	1,89%
0.71	13	208	4	57	282	70	78,37%	76,47%	18,57%	1,89%
0.72	13	208	4	57	282	70	78,37%	76,47%	18,57%	1,89%
0.73	13	208	4	57	282	70	78,37%	76,47%	18,57%	1,89%
0.74	13	208	4	57	282	70	78,37%	76,47%	18,57%	1,89%
0.75	13	208	4	57	282	70	78,37%	76,47%	18,57%	1,89%
0.76	13	208	4	57	282	70	78,37%	76,47%	18,57%	1,89%
0.77	13	208	4	57	282	70	78,37%	76,47%	18,57%	1,89%
0.78	13	208	4	57	282	70	78,37%	76,47%	18,57%	1,89%
0.79	13	208	4	57	282	70	78,37%	76,47%	18,57%	1,89%
0.8	13	208	4	57	282	70	78,37%	76,47%	18,57%	1,89%
0.81	13	208	4	57	282	70	78,37%	76,47%	18,57%	1,89%
0.82	13	208	4	57	282	70	78,37%	76,47%	18,57%	1,89%
0.83	13	208	4	57	282	70	78,37%	76,47%	18,57%	1,89%
0.84	13	208	4	57	282	70	78,37%	76,47%	18,57%	1,89%
0.85	13	208	4	57	282	70	78,37%	76,47%	18,57%	1,89%
0.86	10	208	4	60	282	70	77,30%	71,43%	14,29%	1,89%
0.87	10	208	4	60	282	70	77,30%	71,43%	14,29%	1,89%
0.88	6	209	3	64	282	70	76,24%	66,67%	8,57%	1,42%
0.89	3	209	3	67	282	70	75,18%	50,00%	4,29%	1,42%
0.9	3	209	3	67	282	70	75,18%	50,00%	4,29%	1,42%
0.91	3	209	3	67	282	70	75,18%	50,00%	4,29%	1,42%
0.92	3	209	3	67	282	70	75,18%	50,00%	4,29%	1,42%
0.93	3	209	3	67	282	70	75,18%	50,00%	4,29%	1,42%
0.94	3	210	2	67	282	70	75,53%	60,00%	4,29%	0,94%
0.95	3	210	2	67	282	70	75,53%	60,00%	4,29%	0,94%
0.96	3	210	2	67	282	70	75,53%	60,00%	4,29%	0,94%
0.97	3	210	2	67	282	70	75,53%	60,00%	4,29%	0,94%
0.98	3	210	2	67	282	70	75,53%	60,00%	4,29%	0,94%
0.99	3	210	2	67	282	70	75,53%	60,00%	4,29%	0,94%
1.00	0	212	0	70	282	70	75,18%	100.00%	0,00%	0,00%

6.1.4.4 Naive Bayes

para	TP	TN	FP	FN	Total	Total Attachments	Accuracy	Precision	Recall	Fallout
0.0	70	0	212	0	282	70	24,82%	24,82%	100,00%	100,00%
0.01	70	0	212	0	282	70	24,82%	24,82%	100,00%	100,00%
0.02	70	1	211	0	282	70	25,18%	24,91%	100,00%	99,53%
0.03	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.04	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.05	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.06	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.07	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.08	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.09	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.1	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.11	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.12	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.13	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.14	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.15	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.16	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.17	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.18	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.19	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.2	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.21	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.22	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.23	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.24	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.25	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.26	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.27	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.28	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.29	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.3	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.31	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.32	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.33	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.34	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.35	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.36	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.37	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.38	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.39	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.4	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.41	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.42	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.43	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.44	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.45	70	7	205	0	282	70	27,30%	25,45%	100,00%	96,70%
0.46	70	10	202	0	282	70	28,37%	25,74%	100,00%	95,28%
0.47	50	29	183	20	282	70	28,01%	21,46%	71,43%	86,32%
0.48	50	56	156	20	282	70	37,59%	24,27%	71,43%	73,58%
0.49	45	100	112	25	282	70	51,42%	28,66%	64,29%	52,83%
0.5	36	184	28	34	282	70	78,01%	56,25%	51,43%	13,21%
0.51	22	204	8	48	282	70	80,14%	73,33%	31,43%	3,77%
0.52	16	207	5	54	282	70	79,08%	76,19%	22,86%	2,36%
0.53	2	210	2	68	282	70	75,18%	50,00%	2,86%	0,94%
0.54	1	212	0	69	282	70	75,53%	100,00%	1,43%	0,00%
0.55	1	212	0	69	282	70	75,53%	100,00%	1,43%	0,00%
0.56	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.57	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.58	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.59	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%

para	TP	TN	FP	FN	Total	Total Attachments	Accuracy	Precision	Recall	Fallout
0.6	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.61	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.62	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.63	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.64	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.65	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.66	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.67	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.68	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.69	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.7	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.71	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.72	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.73	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.74	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.75	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.76	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.77	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.78	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.79	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.8	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.81	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.82	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.83	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.84	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.85	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.86	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.87	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.88	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.89	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.9	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.91	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.92	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.93	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.94	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.95	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.96	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.97	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.98	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.99	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
1.00	0	212	0	70	282	70	75,18%	100.00%	0,00%	0,00%

6.1.4.5 Combined

para	TP	TN	FP	FN	Total	Total Attachments	Accuracy	Precision	Recall	Fallout
0.0	70	0	212	0	282	70	24,82%	24,82%	100,00%	100,00%
0.01	70	0	212	0	282	70	24,82%	24,82%	100,00%	100,00%
0.02	70	0	212	0	282	70	24,82%	24,82%	100,00%	100,00%
0.03	70	0	212	0	282	70	24,82%	24,82%	100,00%	100,00%
0.04	70	0	212	0	282	70	24,82%	24,82%	100,00%	100,00%
0.05	70	0	212	0	282	70	24,82%	24,82%	100,00%	100,00%
0.06	70	0	212	0	282	70	24,82%	24,82%	100,00%	100,00%
0.07	70	0	212	0	282	70	24,82%	24,82%	100,00%	100,00%
0.08	70	0	212	0	282	70	24,82%	24,82%	100,00%	100,00%
0.09	70	0	212	0	282	70	24,82%	24,82%	100,00%	100,00%
0.1	70	0	212	0	282	70	24,82%	24,82%	100,00%	100,00%
0.11	70	0	212	0	282	70	24,82%	24,82%	100,00%	100,00%
0.12	70	0	212	0	282	70	24,82%	24,82%	100,00%	100,00%
0.13	70	0	212	0	282	70	24,82%	24,82%	100,00%	100,00%
0.14	70	0	212	0	282	70	24,82%	24,82%	100,00%	100,00%
0.15	70	0	212	0	282	70	24,82%	24,82%	100,00%	100,00%
0.16	70	0	212	0	282	70	24,82%	24,82%	100,00%	100,00%
0.17	70	0	212	0	282	70	24,82%	24,82%	100,00%	100,00%
0.18	70	0	212	0	282	70	24,82%	24,82%	100,00%	100,00%
0.19	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.2	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.21	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.22	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.23	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.24	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.25	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.26	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.27	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.28	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.29	70	3	209	0	282	70	25,89%	25,09%	100,00%	98,58%
0.3	70	6	206	0	282	70	26,95%	25,36%	100,00%	97,17%
0.31	70	16	196	0	282	70	30,50%	26,32%	100,00%	92,45%
0.32	70	54	158	0	282	70	43,97%	30,70%	100,00%	74,53%
0.33	69	135	77	1	282	70	72,34%	47,26%	98,57%	36,32%
0.34	68	200	12	2	282	70	95,04%	85,00%	97,14%	5,66%
0.35	65	204	8	5	282	70	95,39%	89,04%	92,86%	3,77%
0.36	42	206	6	28	282	70	87,94%	87,50%	60,00%	2,83%
0.37	34	206	6	36	282	70	85,11%	85,00%	48,57%	2,83%
0.38	34	206	6	36	282	70	85,11%	85,00%	48,57%	2,83%
0.39	34	206	6	36	282	70	85,11%	85,00%	48,57%	2,83%
0.4	34	206	6	36	282	70	85,11%	85,00%	48,57%	2,83%
0.41	34	206	6	36	282	70	85,11%	85,00%	48,57%	2,83%
0.42	34	206	6	36	282	70	85,11%	85,00%	48,57%	2,83%
0.43	34	206	6	36	282	70	85,11%	85,00%	48,57%	2,83%
0.44	34	206	6	36	282	70	85,11%	85,00%	48,57%	2,83%
0.45	34	207	5	36	282	70	85,46%	87,18%	48,57%	2,36%
0.46	34	208	4	36	282	70	85,82%	89,47%	48,57%	1,89%
0.47	34	208	4	36	282	70	85,82%	89,47%	48,57%	1,89%
0.48	34	208	4	36	282	70	85,82%	89,47%	48,57%	1,89%
0.49	30	208	4	40	282	70	84,40%	88,24%	42,86%	1,89%
0.5	14	210	2	56	282	70	79,43%	87,50%	20,00%	0,94%
0.51	14	210	2	56	282	70	79,43%	87,50%	20,00%	0,94%
0.52	14	210	2	56	282	70	79,43%	87,50%	20,00%	0,94%
0.53	13	210	2	57	282	70	79,08%	86,67%	18,57%	0,94%
0.54	13	210	2	57	282	70	79,08%	86,67%	18,57%	0,94%
0.55	13	210	2	57	282	70	79,08%	86,67%	18,57%	0,94%
0.56	13	210	2	57	282	70	79,08%	86,67%	18,57%	0,94%
0.57	13	210	2	57	282	70	79,08%	86,67%	18,57%	0,94%
0.58	13	210	2	57	282	70	79,08%	86,67%	18,57%	0,94%
0.59	13	210	2	57	282	70	79,08%	86,67%	18,57%	0,94%

para	TP	TN	FP	FN	Total	Total Attachments	Accuracy	Precision	Recall	Fallout
0.6	13	210	2	57	282	70	79,08%	86,67%	18,57%	0,94%
0.61	13	210	2	57	282	70	79,08%	86,67%	18,57%	0,94%
0.62	13	210	2	57	282	70	79,08%	86,67%	18,57%	0,94%
0.63	13	210	2	57	282	70	79,08%	86,67%	18,57%	0,94%
0.64	9	210	2	61	282	70	77,66%	81,82%	12,86%	0,94%
0.65	4	210	2	66	282	70	75,89%	66,67%	5,71%	0,94%
0.66	4	210	2	66	282	70	75,89%	66,67%	5,71%	0,94%
0.67	4	211	1	66	282	70	76,24%	80,00%	5,71%	0,47%
0.68	2	211	1	68	282	70	75,53%	66,67%	2,86%	0,47%
0.69	2	212	0	68	282	70	75,89%	100,00%	2,86%	0,00%
0.7	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.71	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.72	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.73	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.74	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.75	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.76	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.77	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.78	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.79	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.8	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.81	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.82	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.83	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.84	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.85	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.86	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.87	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.88	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.89	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.9	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.91	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.92	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.93	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.94	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.95	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.96	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.97	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.98	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
0.99	0	212	0	70	282	70	75,18%	100,00%	0,00%	0,00%
1.00	0	212	0	70	282	70	75,18%	100.00%	0,00%	0,00%

6.1.5 Datensatz 5

6.1.5.1 Einfacher Algorithmus

para	TP	TN	FP	FN	Total	Total Attachments	Accuracy	Precision	Recall	Fallout
1	43	715	111	53	922	96	82,21%	27,92%	44,79%	13,44%

6.1.5.2 Erweiterter Algorithmus

para	TP	TN	FP	FN	Total	Total Attachments	Accuracy	Precision	Recall	Fallout
2	51	724	102	45	922	96	84,06%	33,33%	53,13%	12,35%
3	46	782	44	50	922	96	89,80%	51,11%	47,92%	5,33%
4	49	782	44	47	922	96	90,13%	52,69%	51,04%	5,33%
5	45	798	28	51	922	96	91,43%	61,64%	46,88%	3,39%
6	42	803	23	54	922	96	91,65%	64,62%	43,75%	2,78%
7	42	806	20	54	922	96	91,97%	67,74%	43,75%	2,42%
8	42	808	18	54	922	96	92,19%	70,00%	43,75%	2,18%
9	42	807	19	54	922	96	92,08%	68,85%	43,75%	2,30%
10	41	808	18	55	922	96	92,08%	69,49%	42,71%	2,18%
11	42	808	18	54	922	96	92,19%	70,00%	43,75%	2,18%
12	41	809	17	55	922	96	92,19%	70,69%	42,71%	2,06%
13	41	806	20	55	922	96	91,87%	67,21%	42,71%	2,42%
14	41	808	18	55	922	96	92,08%	69,49%	42,71%	2,18%
15	38	809	17	58	922	96	91,87%	69,09%	39,58%	2,06%

6.1.5.3 Paul Graham

para	TP	TN	FP	FN	Total	Total Attachments	Accuracy	Precision	Recall	Fallout
0.0	96	0	826	0	922	96	10,41%	10,41%	100,00%	100,00%
0.01	31	807	19	65	922	96	90,89%	62,00%	32,29%	2,30%
0.02	30	810	16	66	922	96	91,11%	65,22%	31,25%	1,94%
0.03	29	812	14	67	922	96	91,21%	67,44%	30,21%	1,69%
0.04	27	812	14	69	922	96	91,00%	65,85%	28,13%	1,69%
0.05	27	812	14	69	922	96	91,00%	65,85%	28,13%	1,69%
0.06	27	812	14	69	922	96	91,00%	65,85%	28,13%	1,69%
0.07	26	812	14	70	922	96	90,89%	65,00%	27,08%	1,69%
0.08	26	812	14	70	922	96	90,89%	65,00%	27,08%	1,69%
0.09	25	812	14	71	922	96	90,78%	64,10%	26,04%	1,69%
0.1	24	812	14	72	922	96	90,67%	63,16%	25,00%	1,69%
0.11	24	812	14	72	922	96	90,67%	63,16%	25,00%	1,69%
0.12	24	812	14	72	922	96	90,67%	63,16%	25,00%	1,69%
0.13	24	813	13	72	922	96	90,78%	64,86%	25,00%	1,57%
0.14	24	814	12	72	922	96	90,89%	66,67%	25,00%	1,45%
0.15	24	814	12	72	922	96	90,89%	66,67%	25,00%	1,45%
0.16	24	814	12	72	922	96	90,89%	66,67%	25,00%	1,45%
0.17	22	814	12	74	922	96	90,67%	64,71%	22,92%	1,45%
0.18	22	814	12	74	922	96	90,67%	64,71%	22,92%	1,45%
0.19	22	814	12	74	922	96	90,67%	64,71%	22,92%	1,45%
0.2	22	814	12	74	922	96	90,67%	64,71%	22,92%	1,45%
0.21	21	814	12	75	922	96	90,56%	63,64%	21,88%	1,45%
0.22	20	814	12	76	922	96	90,46%	62,50%	20,83%	1,45%
0.23	20	815	11	76	922	96	90,56%	64,52%	20,83%	1,33%
0.24	19	815	11	77	922	96	90,46%	63,33%	19,79%	1,33%
0.25	19	815	11	77	922	96	90,46%	63,33%	19,79%	1,33%
0.26	19	815	11	77	922	96	90,46%	63,33%	19,79%	1,33%
0.27	19	815	11	77	922	96	90,46%	63,33%	19,79%	1,33%
0.28	19	815	11	77	922	96	90,46%	63,33%	19,79%	1,33%
0.29	19	815	11	77	922	96	90,46%	63,33%	19,79%	1,33%
0.3	19	815	11	77	922	96	90,46%	63,33%	19,79%	1,33%
0.31	19	815	11	77	922	96	90,46%	63,33%	19,79%	1,33%
0.32	19	815	11	77	922	96	90,46%	63,33%	19,79%	1,33%
0.33	19	815	11	77	922	96	90,46%	63,33%	19,79%	1,33%
0.34	19	815	11	77	922	96	90,46%	63,33%	19,79%	1,33%
0.35	19	815	11	77	922	96	90,46%	63,33%	19,79%	1,33%
0.36	19	815	11	77	922	96	90,46%	63,33%	19,79%	1,33%
0.37	19	815	11	77	922	96	90,46%	63,33%	19,79%	1,33%
0.38	19	815	11	77	922	96	90,46%	63,33%	19,79%	1,33%
0.39	19	815	11	77	922	96	90,46%	63,33%	19,79%	1,33%
0.4	19	815	11	77	922	96	90,46%	63,33%	19,79%	1,33%
0.41	19	816	10	77	922	96	90,56%	65,52%	19,79%	1,21%
0.42	19	816	10	77	922	96	90,56%	65,52%	19,79%	1,21%
0.43	19	816	10	77	922	96	90,56%	65,52%	19,79%	1,21%
0.44	19	816	10	77	922	96	90,56%	65,52%	19,79%	1,21%
0.45	19	816	10	77	922	96	90,56%	65,52%	19,79%	1,21%
0.46	19	816	10	77	922	96	90,56%	65,52%	19,79%	1,21%
0.47	19	816	10	77	922	96	90,56%	65,52%	19,79%	1,21%
0.48	19	816	10	77	922	96	90,56%	65,52%	19,79%	1,21%
0.49	19	816	10	77	922	96	90,56%	65,52%	19,79%	1,21%
0.5	18	816	10	78	922	96	90,46%	64,29%	18,75%	1,21%
0.51	18	818	8	78	922	96	90,67%	69,23%	18,75%	0,97%
0.52	18	818	8	78	922	96	90,67%	69,23%	18,75%	0,97%
0.53	18	818	8	78	922	96	90,67%	69,23%	18,75%	0,97%
0.54	18	818	8	78	922	96	90,67%	69,23%	18,75%	0,97%
0.55	18	818	8	78	922	96	90,67%	69,23%	18,75%	0,97%
0.56	18	818	8	78	922	96	90,67%	69,23%	18,75%	0,97%
0.57	17	819	7	79	922	96	90,67%	70,83%	17,71%	0,85%
0.58	16	819	7	80	922	96	90,56%	69,57%	16,67%	0,85%
0.59	16	819	7	80	922	96	90,56%	69,57%	16,67%	0,85%

para	TP	TN	FP	FN	Total	Total Attachments	Accuracy	Precision	Recall	Fallout
0.6	16	819	7	80	922	96	90,56%	69,57%	16,67%	0,85%
0.61	15	819	7	81	922	96	90,46%	68,18%	15,63%	0,85%
0.62	15	819	7	81	922	96	90,46%	68,18%	15,63%	0,85%
0.63	15	819	7	81	922	96	90,46%	68,18%	15,63%	0,85%
0.64	15	819	7	81	922	96	90,46%	68,18%	15,63%	0,85%
0.65	15	819	7	81	922	96	90,46%	68,18%	15,63%	0,85%
0.66	15	819	7	81	922	96	90,46%	68,18%	15,63%	0,85%
0.67	15	820	6	81	922	96	90,56%	71,43%	15,63%	0,73%
0.68	15	820	6	81	922	96	90,56%	71,43%	15,63%	0,73%
0.69	15	820	6	81	922	96	90,56%	71,43%	15,63%	0,73%
0.7	9	822	4	87	922	96	90,13%	69,23%	9,38%	0,48%
0.71	9	822	4	87	922	96	90,13%	69,23%	9,38%	0,48%
0.72	9	822	4	87	922	96	90,13%	69,23%	9,38%	0,48%
0.73	9	822	4	87	922	96	90,13%	69,23%	9,38%	0,48%
0.74	8	822	4	88	922	96	90,02%	66,67%	8,33%	0,48%
0.75	8	822	4	88	922	96	90,02%	66,67%	8,33%	0,48%
0.76	8	822	4	88	922	96	90,02%	66,67%	8,33%	0,48%
0.77	8	822	4	88	922	96	90,02%	66,67%	8,33%	0,48%
0.78	8	822	4	88	922	96	90,02%	66,67%	8,33%	0,48%
0.79	8	822	4	88	922	96	90,02%	66,67%	8,33%	0,48%
0.8	8	822	4	88	922	96	90,02%	66,67%	8,33%	0,48%
0.81	8	822	4	88	922	96	90,02%	66,67%	8,33%	0,48%
0.82	8	822	4	88	922	96	90,02%	66,67%	8,33%	0,48%
0.83	8	822	4	88	922	96	90,02%	66,67%	8,33%	0,48%
0.84	8	822	4	88	922	96	90,02%	66,67%	8,33%	0,48%
0.85	8	822	4	88	922	96	90,02%	66,67%	8,33%	0,48%
0.86	8	822	4	88	922	96	90,02%	66,67%	8,33%	0,48%
0.87	6	822	4	90	922	96	89,80%	60,00%	6,25%	0,48%
0.88	6	822	4	90	922	96	89,80%	60,00%	6,25%	0,48%
0.89	6	822	4	90	922	96	89,80%	60,00%	6,25%	0,48%
0.9	6	822	4	90	922	96	89,80%	60,00%	6,25%	0,48%
0.91	6	822	4	90	922	96	89,80%	60,00%	6,25%	0,48%
0.92	6	822	4	90	922	96	89,80%	60,00%	6,25%	0,48%
0.93	6	822	4	90	922	96	89,80%	60,00%	6,25%	0,48%
0.94	6	822	4	90	922	96	89,80%	60,00%	6,25%	0,48%
0.95	6	822	4	90	922	96	89,80%	60,00%	6,25%	0,48%
0.96	6	822	4	90	922	96	89,80%	60,00%	6,25%	0,48%
0.97	6	822	4	90	922	96	89,80%	60,00%	6,25%	0,48%
0.98	5	822	4	91	922	96	89,70%	55,56%	5,21%	0,48%
0.99	3	822	4	93	922	96	89,48%	42,86%	3,13%	0,48%
1.0	0	826	0	96	922	96	89,59%	100,00%	0,00%	0,00%

6.1.5.4 Naive Bayes

para	TP	TN	FP	FN	Total	Total Attachments	Accuracy	Precision	Recall	Fallout
0.0	96	0	826	0	922	96	10,41%	10,41%	100,00%	100,00%
0.01	96	0	826	0	922	96	10,41%	10,41%	100,00%	100,00%
0.02	96	1	825	0	922	96	10,52%	10,42%	100,00%	99,88%
0.03	96	2	824	0	922	96	10,63%	10,43%	100,00%	99,76%
0.04	96	2	824	0	922	96	10,63%	10,43%	100,00%	99,76%
0.05	96	2	824	0	922	96	10,63%	10,43%	100,00%	99,76%
0.06	96	2	824	0	922	96	10,63%	10,43%	100,00%	99,76%
0.07	96	2	824	0	922	96	10,63%	10,43%	100,00%	99,76%
0.08	96	2	824	0	922	96	10,63%	10,43%	100,00%	99,76%
0.09	96	2	824	0	922	96	10,63%	10,43%	100,00%	99,76%
0.1	96	2	824	0	922	96	10,63%	10,43%	100,00%	99,76%
0.11	96	2	824	0	922	96	10,63%	10,43%	100,00%	99,76%
0.12	96	2	824	0	922	96	10,63%	10,43%	100,00%	99,76%
0.13	96	2	824	0	922	96	10,63%	10,43%	100,00%	99,76%
0.14	96	2	824	0	922	96	10,63%	10,43%	100,00%	99,76%
0.15	96	2	824	0	922	96	10,63%	10,43%	100,00%	99,76%
0.16	96	2	824	0	922	96	10,63%	10,43%	100,00%	99,76%
0.17	96	2	824	0	922	96	10,63%	10,43%	100,00%	99,76%
0.18	96	2	824	0	922	96	10,63%	10,43%	100,00%	99,76%
0.19	96	2	824	0	922	96	10,63%	10,43%	100,00%	99,76%
0.2	96	2	824	0	922	96	10,63%	10,43%	100,00%	99,76%
0.21	96	2	824	0	922	96	10,63%	10,43%	100,00%	99,76%
0.22	96	2	824	0	922	96	10,63%	10,43%	100,00%	99,76%
0.23	96	2	824	0	922	96	10,63%	10,43%	100,00%	99,76%
0.24	96	2	824	0	922	96	10,63%	10,43%	100,00%	99,76%
0.25	96	2	824	0	922	96	10,63%	10,43%	100,00%	99,76%
0.26	96	2	824	0	922	96	10,63%	10,43%	100,00%	99,76%
0.27	96	2	824	0	922	96	10,63%	10,43%	100,00%	99,76%
0.28	96	2	824	0	922	96	10,63%	10,43%	100,00%	99,76%
0.29	96	2	824	0	922	96	10,63%	10,43%	100,00%	99,76%
0.3	96	2	824	0	922	96	10,63%	10,43%	100,00%	99,76%
0.31	96	2	824	0	922	96	10,63%	10,43%	100,00%	99,76%
0.32	96	2	824	0	922	96	10,63%	10,43%	100,00%	99,76%
0.33	96	2	824	0	922	96	10,63%	10,43%	100,00%	99,76%
0.34	96	2	824	0	922	96	10,63%	10,43%	100,00%	99,76%
0.35	96	2	824	0	922	96	10,63%	10,43%	100,00%	99,76%
0.36	96	2	824	0	922	96	10,63%	10,43%	100,00%	99,76%
0.37	96	2	824	0	922	96	10,63%	10,43%	100,00%	99,76%
0.38	96	2	824	0	922	96	10,63%	10,43%	100,00%	99,76%
0.39	96	2	824	0	922	96	10,63%	10,43%	100,00%	99,76%
0.4	96	5	821	0	922	96	10,95%	10,47%	100,00%	99,39%
0.41	96	5	821	0	922	96	10,95%	10,47%	100,00%	99,39%
0.42	96	5	821	0	922	96	10,95%	10,47%	100,00%	99,39%
0.43	96	5	821	0	922	96	10,95%	10,47%	100,00%	99,39%
0.44	96	7	819	0	922	96	11,17%	10,49%	100,00%	99,15%
0.45	96	9	817	0	922	96	11,39%	10,51%	100,00%	98,91%
0.46	91	18	808	5	922	96	11,82%	10,12%	94,79%	97,82%
0.47	90	73	753	6	922	96	17,68%	10,68%	93,75%	91,16%
0.48	85	269	557	11	922	96	38,39%	13,24%	88,54%	67,43%
0.49	74	618	208	22	922	96	75,05%	26,24%	77,08%	25,18%
0.5	37	800	26	59	922	96	90,78%	58,73%	38,54%	3,15%
0.51	21	815	11	75	922	96	90,67%	65,63%	21,88%	1,33%
0.52	7	822	4	89	922	96	89,91%	63,64%	7,29%	0,48%
0.53	1	826	0	95	922	96	89,70%	100,00%	1,04%	0,00%
0.54	1	826	0	95	922	96	89,70%	100,00%	1,04%	0,00%
0.55	0	826	0	96	922	96	89,59%	100,00%	0,00%	0,00%
0.56	0	826	0	96	922	96	89,59%	100,00%	0,00%	0,00%
0.57	0	826	0	96	922	96	89,59%	100,00%	0,00%	0,00%
0.58	0	826	0	96	922	96	89,59%	100,00%	0,00%	0,00%
0.59	0	826	0	96	922	96	89,59%	100,00%	0,00%	0,00%

para	TP	TN	FP	FN	Total	Total Attachments	Accuracy	Precision	Recall	Fallout
0.6	16	819	7	80	922	96	90,56%	69,57%	16,67%	0,85%
0.61	15	819	7	81	922	96	90,46%	68,18%	15,63%	0,85%
0.62	15	819	7	81	922	96	90,46%	68,18%	15,63%	0,85%
0.63	15	819	7	81	922	96	90,46%	68,18%	15,63%	0,85%
0.64	15	819	7	81	922	96	90,46%	68,18%	15,63%	0,85%
0.65	15	819	7	81	922	96	90,46%	68,18%	15,63%	0,85%
0.66	15	819	7	81	922	96	90,46%	68,18%	15,63%	0,85%
0.67	15	820	6	81	922	96	90,56%	71,43%	15,63%	0,73%
0.68	15	820	6	81	922	96	90,56%	71,43%	15,63%	0,73%
0.69	15	820	6	81	922	96	90,56%	71,43%	15,63%	0,73%
0.7	9	822	4	87	922	96	90,13%	69,23%	9,38%	0,48%
0.71	9	822	4	87	922	96	90,13%	69,23%	9,38%	0,48%
0.72	9	822	4	87	922	96	90,13%	69,23%	9,38%	0,48%
0.73	9	822	4	87	922	96	90,13%	69,23%	9,38%	0,48%
0.74	8	822	4	88	922	96	90,02%	66,67%	8,33%	0,48%
0.75	8	822	4	88	922	96	90,02%	66,67%	8,33%	0,48%
0.76	8	822	4	88	922	96	90,02%	66,67%	8,33%	0,48%
0.77	8	822	4	88	922	96	90,02%	66,67%	8,33%	0,48%
0.78	8	822	4	88	922	96	90,02%	66,67%	8,33%	0,48%
0.79	8	822	4	88	922	96	90,02%	66,67%	8,33%	0,48%
0.8	8	822	4	88	922	96	90,02%	66,67%	8,33%	0,48%
0.81	8	822	4	88	922	96	90,02%	66,67%	8,33%	0,48%
0.82	8	822	4	88	922	96	90,02%	66,67%	8,33%	0,48%
0.83	8	822	4	88	922	96	90,02%	66,67%	8,33%	0,48%
0.84	8	822	4	88	922	96	90,02%	66,67%	8,33%	0,48%
0.85	8	822	4	88	922	96	90,02%	66,67%	8,33%	0,48%
0.86	8	822	4	88	922	96	90,02%	66,67%	8,33%	0,48%
0.87	6	822	4	90	922	96	89,80%	60,00%	6,25%	0,48%
0.88	6	822	4	90	922	96	89,80%	60,00%	6,25%	0,48%
0.89	6	822	4	90	922	96	89,80%	60,00%	6,25%	0,48%
0.9	6	822	4	90	922	96	89,80%	60,00%	6,25%	0,48%
0.91	6	822	4	90	922	96	89,80%	60,00%	6,25%	0,48%
0.92	6	822	4	90	922	96	89,80%	60,00%	6,25%	0,48%
0.93	6	822	4	90	922	96	89,80%	60,00%	6,25%	0,48%
0.94	6	822	4	90	922	96	89,80%	60,00%	6,25%	0,48%
0.95	6	822	4	90	922	96	89,80%	60,00%	6,25%	0,48%
0.96	6	822	4	90	922	96	89,80%	60,00%	6,25%	0,48%
0.97	6	822	4	90	922	96	89,80%	60,00%	6,25%	0,48%
0.98	5	822	4	91	922	96	89,70%	55,56%	5,21%	0,48%
0.99	3	822	4	93	922	96	89,48%	42,86%	3,13%	0,48%
1.0	0	826	0	96	922	96	89,59%	100,00%	0,00%	0,00%

6.1.5.5 Combined

para	TP	TN	FP	FN	Total	Total Attachments	Accuracy	Precision	Recall	Fallout
0.0	96	0	826	0	922	96	10,41%	10,41%	100,00%	100,00%
0.01	96	0	826	0	922	96	10,41%	10,41%	100,00%	100,00%
0.02	96	0	826	0	922	96	10,41%	10,41%	100,00%	100,00%
0.03	96	0	826	0	922	96	10,41%	10,41%	100,00%	100,00%
0.04	96	0	826	0	922	96	10,41%	10,41%	100,00%	100,00%
0.05	96	0	826	0	922	96	10,41%	10,41%	100,00%	100,00%
0.06	96	0	826	0	922	96	10,41%	10,41%	100,00%	100,00%
0.07	96	0	826	0	922	96	10,41%	10,41%	100,00%	100,00%
0.08	96	0	826	0	922	96	10,41%	10,41%	100,00%	100,00%
0.09	96	0	826	0	922	96	10,41%	10,41%	100,00%	100,00%
0.1	96	0	826	0	922	96	10,41%	10,41%	100,00%	100,00%
0.11	96	0	826	0	922	96	10,41%	10,41%	100,00%	100,00%
0.12	96	0	826	0	922	96	10,41%	10,41%	100,00%	100,00%
0.13	96	0	826	0	922	96	10,41%	10,41%	100,00%	100,00%
0.14	96	0	826	0	922	96	10,41%	10,41%	100,00%	100,00%
0.15	96	0	826	0	922	96	10,41%	10,41%	100,00%	100,00%
0.16	96	0	826	0	922	96	10,41%	10,41%	100,00%	100,00%
0.17	96	0	826	0	922	96	10,41%	10,41%	100,00%	100,00%
0.18	96	0	826	0	922	96	10,41%	10,41%	100,00%	100,00%
0.19	96	2	824	0	922	96	10,63%	10,43%	100,00%	99,76%
0.2	96	2	824	0	922	96	10,63%	10,43%	100,00%	99,76%
0.21	96	2	824	0	922	96	10,63%	10,43%	100,00%	99,76%
0.22	96	2	824	0	922	96	10,63%	10,43%	100,00%	99,76%
0.23	96	2	824	0	922	96	10,63%	10,43%	100,00%	99,76%
0.24	96	2	824	0	922	96	10,63%	10,43%	100,00%	99,76%
0.25	96	2	824	0	922	96	10,63%	10,43%	100,00%	99,76%
0.26	96	2	824	0	922	96	10,63%	10,43%	100,00%	99,76%
0.27	96	4	822	0	922	96	10,85%	10,46%	100,00%	99,52%
0.28	96	4	822	0	922	96	10,85%	10,46%	100,00%	99,52%
0.29	96	5	821	0	922	96	10,95%	10,47%	100,00%	99,39%
0.3	96	8	818	0	922	96	11,28%	10,50%	100,00%	99,03%
0.31	96	33	793	0	922	96	13,99%	10,80%	100,00%	96,00%
0.32	96	251	575	0	922	96	37,64%	14,31%	100,00%	69,61%
0.33	91	733	93	5	922	96	89,37%	49,46%	94,79%	11,26%
0.34	77	809	17	19	922	96	96,10%	81,91%	80,21%	2,06%
0.35	44	812	14	52	922	96	92,84%	75,86%	45,83%	1,69%
0.36	37	813	13	59	922	96	92,19%	74,00%	38,54%	1,57%
0.37	31	814	12	65	922	96	91,65%	72,09%	32,29%	1,45%
0.38	29	814	12	67	922	96	91,43%	70,73%	30,21%	1,45%
0.39	28	814	12	68	922	96	91,32%	70,00%	29,17%	1,45%
0.4	28	815	11	68	922	96	91,43%	71,79%	29,17%	1,33%
0.41	27	815	11	69	922	96	91,32%	71,05%	28,13%	1,33%
0.42	26	815	11	70	922	96	91,21%	70,27%	27,08%	1,33%
0.43	24	815	11	72	922	96	91,00%	68,57%	25,00%	1,33%
0.44	23	816	10	73	922	96	91,00%	69,70%	23,96%	1,21%
0.45	23	816	10	73	922	96	91,00%	69,70%	23,96%	1,21%
0.46	23	816	10	73	922	96	91,00%	69,70%	23,96%	1,21%
0.47	22	817	9	74	922	96	91,00%	70,97%	22,92%	1,09%
0.48	22	817	9	74	922	96	91,00%	70,97%	22,92%	1,09%
0.49	22	818	8	74	922	96	91,11%	73,33%	22,92%	0,97%
0.5	22	818	8	74	922	96	91,11%	73,33%	22,92%	0,97%
0.51	21	818	8	75	922	96	91,00%	72,41%	21,88%	0,97%
0.52	20	818	8	76	922	96	90,89%	71,43%	20,83%	0,97%
0.53	19	818	8	77	922	96	90,78%	70,37%	19,79%	0,97%
0.54	14	820	6	82	922	96	90,46%	70,00%	14,58%	0,73%
0.55	13	820	6	83	922	96	90,35%	68,42%	13,54%	0,73%
0.56	12	820	6	84	922	96	90,24%	66,67%	12,50%	0,73%
0.57	12	821	5	84	922	96	90,35%	70,59%	12,50%	0,61%
0.58	12	821	5	84	922	96	90,35%	70,59%	12,50%	0,61%
0.59	11	821	5	85	922	96	90,24%	68,75%	11,46%	0,61%

para	TP	TN	FP	FN	Total	Total Attachments	Accuracy	Precision	Recall	Fallout
0.6	11	821	5	85	922	96	90,24%	68,75%	11,46%	0,61%
0.61	11	822	4	85	922	96	90,35%	73,33%	11,46%	0,48%
0.62	11	822	4	85	922	96	90,35%	73,33%	11,46%	0,48%
0.63	11	822	4	85	922	96	90,35%	73,33%	11,46%	0,48%
0.64	9	822	4	87	922	96	90,13%	69,23%	9,38%	0,48%
0.65	9	822	4	87	922	96	90,13%	69,23%	9,38%	0,48%
0.66	8	822	4	88	922	96	90,02%	66,67%	8,33%	0,48%
0.67	7	822	4	89	922	96	89,91%	63,64%	7,29%	0,48%
0.68	5	822	4	91	922	96	89,70%	55,56%	5,21%	0,48%
0.69	1	825	1	95	922	96	89,59%	50,00%	1,04%	0,12%
0.7	1	826	0	95	922	96	89,70%	100,00%	1,04%	0,00%
0.71	1	826	0	95	922	96	89,70%	100,00%	1,04%	0,00%
0.72	0	826	0	96	922	96	89,59%	100,00%	0,00%	0,00%
0.73	0	826	0	96	922	96	89,59%	100,00%	0,00%	0,00%
0.74	0	826	0	96	922	96	89,59%	100,00%	0,00%	0,00%
0.75	0	826	0	96	922	96	89,59%	100,00%	0,00%	0,00%
0.76	0	826	0	96	922	96	89,59%	100,00%	0,00%	0,00%
0.77	0	826	0	96	922	96	89,59%	100,00%	0,00%	0,00%
0.78	0	826	0	96	922	96	89,59%	100,00%	0,00%	0,00%
0.79	0	826	0	96	922	96	89,59%	100,00%	0,00%	0,00%
0.8	0	826	0	96	922	96	89,59%	100,00%	0,00%	0,00%
0.81	0	826	0	96	922	96	89,59%	100,00%	0,00%	0,00%
0.82	0	826	0	96	922	96	89,59%	100,00%	0,00%	0,00%
0.83	0	826	0	96	922	96	89,59%	100,00%	0,00%	0,00%
0.84	0	826	0	96	922	96	89,59%	100,00%	0,00%	0,00%
0.85	0	826	0	96	922	96	89,59%	100,00%	0,00%	0,00%
0.86	0	826	0	96	922	96	89,59%	100,00%	0,00%	0,00%
0.87	0	826	0	96	922	96	89,59%	100,00%	0,00%	0,00%
0.88	0	826	0	96	922	96	89,59%	100,00%	0,00%	0,00%
0.89	0	826	0	96	922	96	89,59%	100,00%	0,00%	0,00%
0.9	0	826	0	96	922	96	89,59%	100,00%	0,00%	0,00%
0.91	0	826	0	96	922	96	89,59%	100,00%	0,00%	0,00%
0.92	0	826	0	96	922	96	89,59%	100,00%	0,00%	0,00%
0.93	0	826	0	96	922	96	89,59%	100,00%	0,00%	0,00%
0.94	0	826	0	96	922	96	89,59%	100,00%	0,00%	0,00%
0.95	0	826	0	96	922	96	89,59%	100,00%	0,00%	0,00%
0.96	0	826	0	96	922	96	89,59%	100,00%	0,00%	0,00%
0.97	0	826	0	96	922	96	89,59%	100,00%	0,00%	0,00%
0.98	0	826	0	96	922	96	89,59%	100,00%	0,00%	0,00%
0.99	0	826	0	96	922	96	89,59%	100,00%	0,00%	0,00%
1.00	0	826	0	96	922	96	89,59%	100,00%	0,00%	0,00%

6.2 Diagramme

6.2.1 Datensatz 1

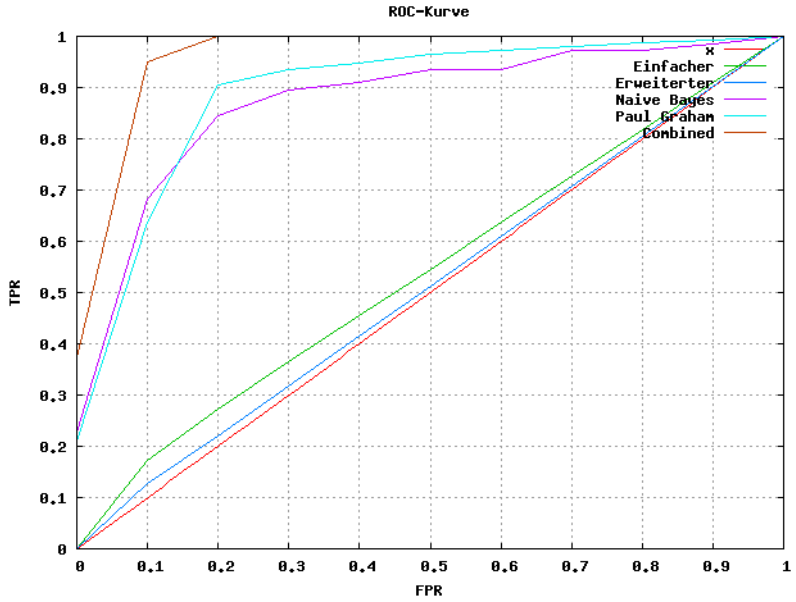

Abbildung 6.1: ROC-Kurven Datensatz 1

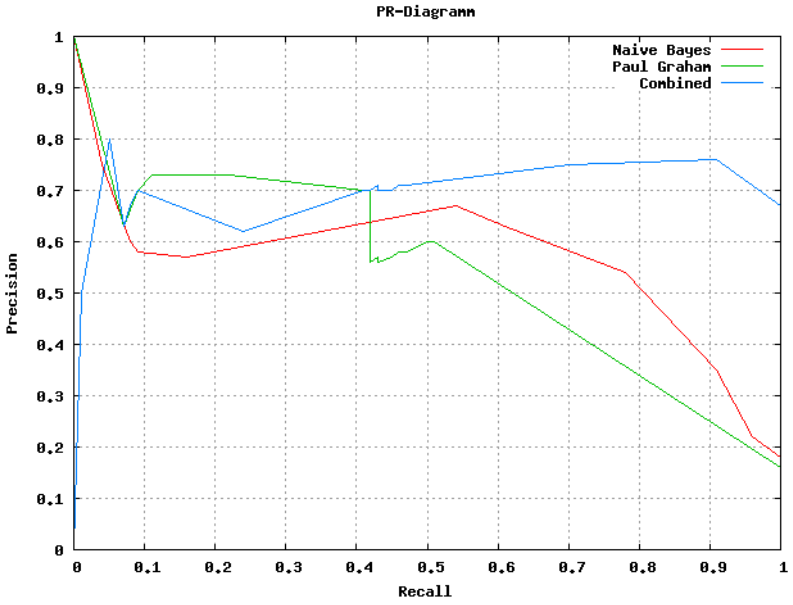

Abbildung 6.2: PR-Diagramm Datensatz 1

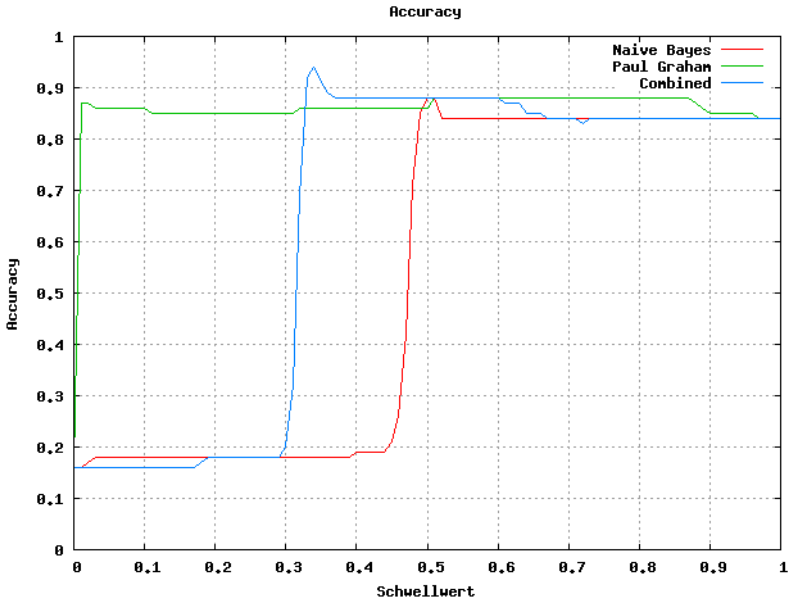

Abbildung 6.3: Accuracy-Diagramm Datensatz 1

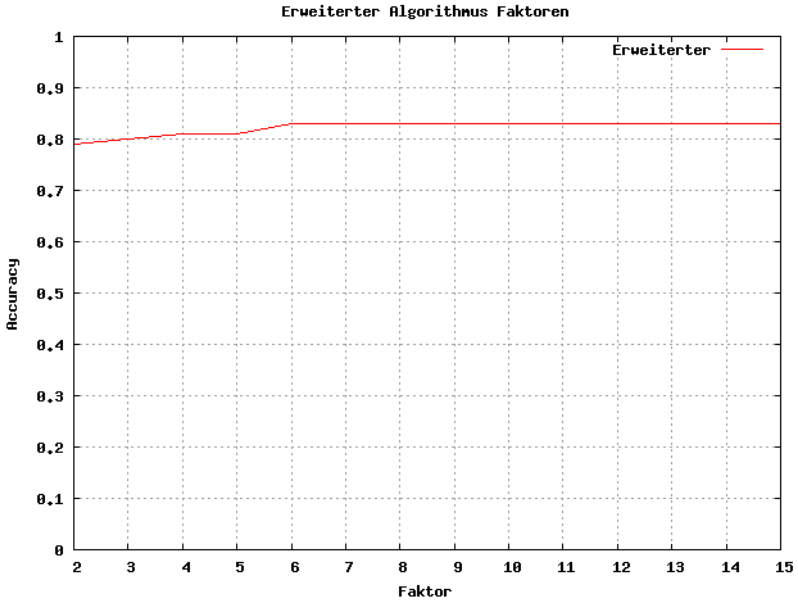

Abbildung 6.4: Erweiterter Algorithmus-Diagramm Datensatz 1

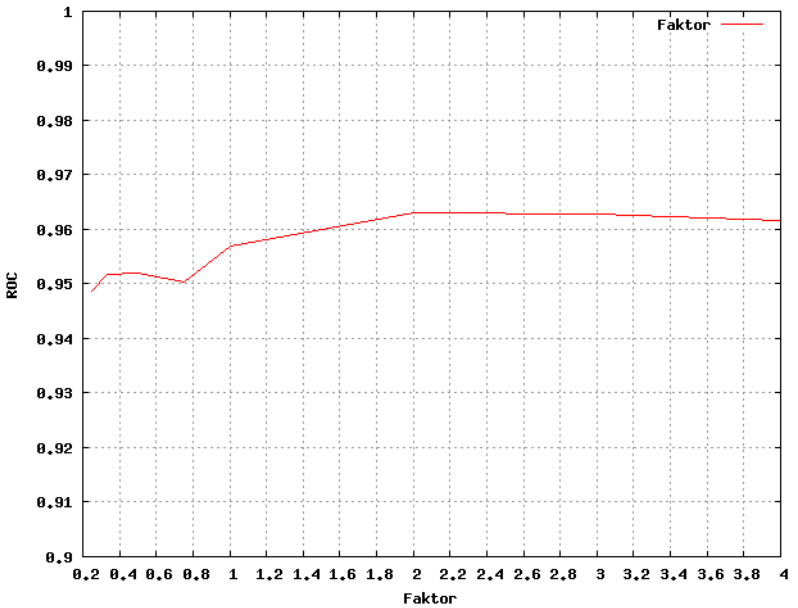

Abbildung 6.5: Combined Parameter ROC Datensatz 1

6.2.2 Datensatz 2

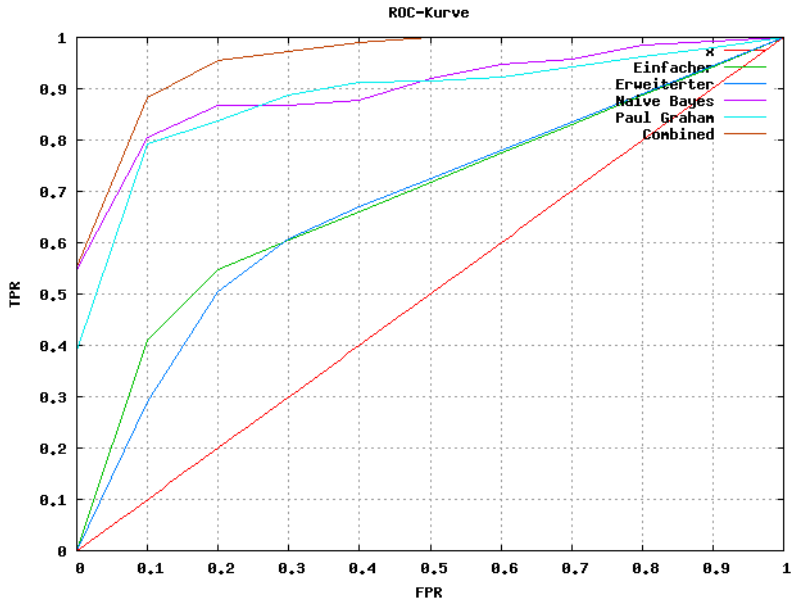

Abbildung 6.6: ROC-Kurven Datensatz 2

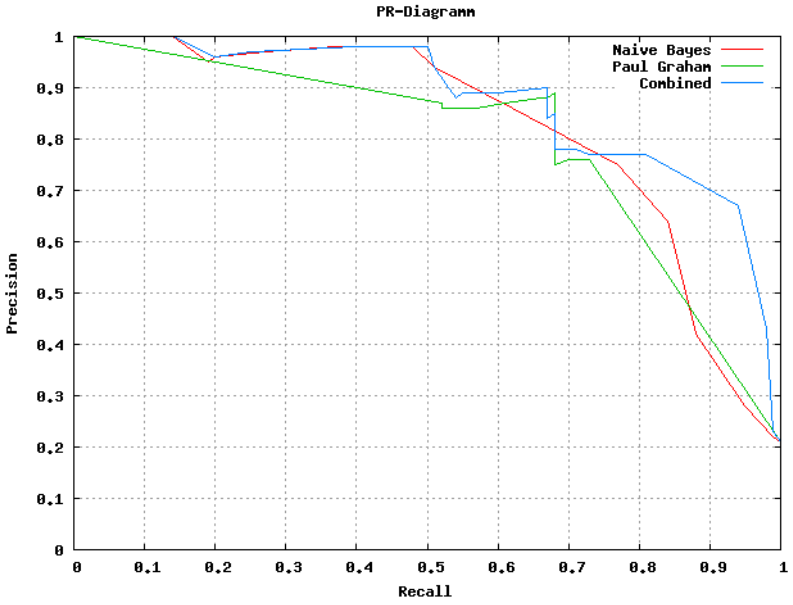

Abbildung 6.7: PR-Diagramm Datensatz 2

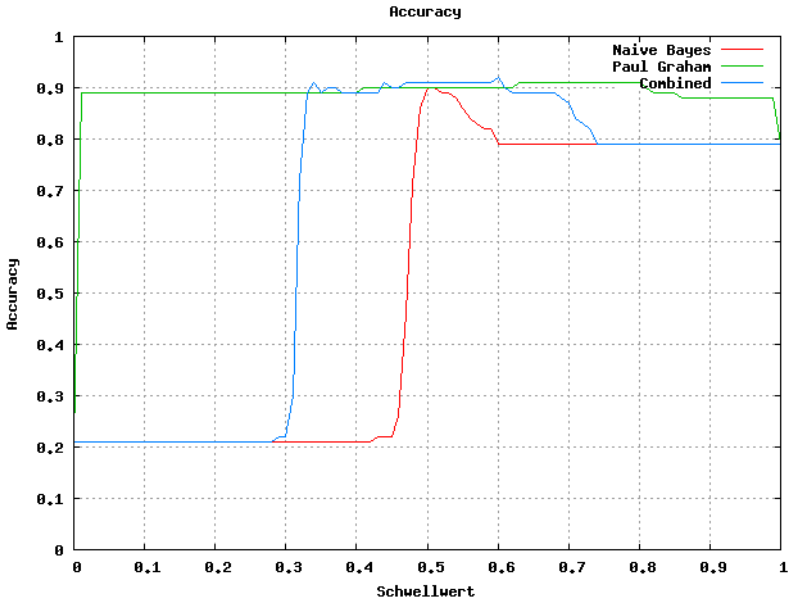

Abbildung 6.8: Accuracy-Diagramm Datensatz 2

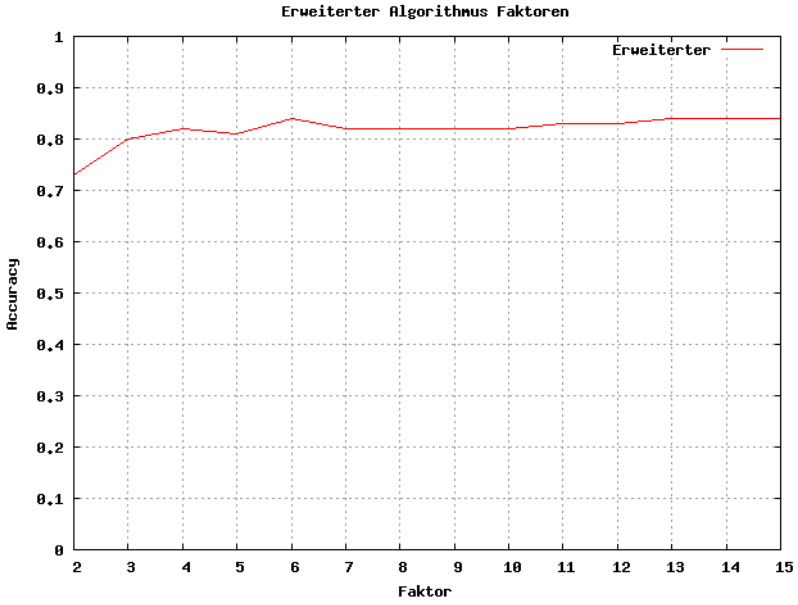

Abbildung 6.9: Erweiterter Algorithmus-Diagramm Datensatz 2

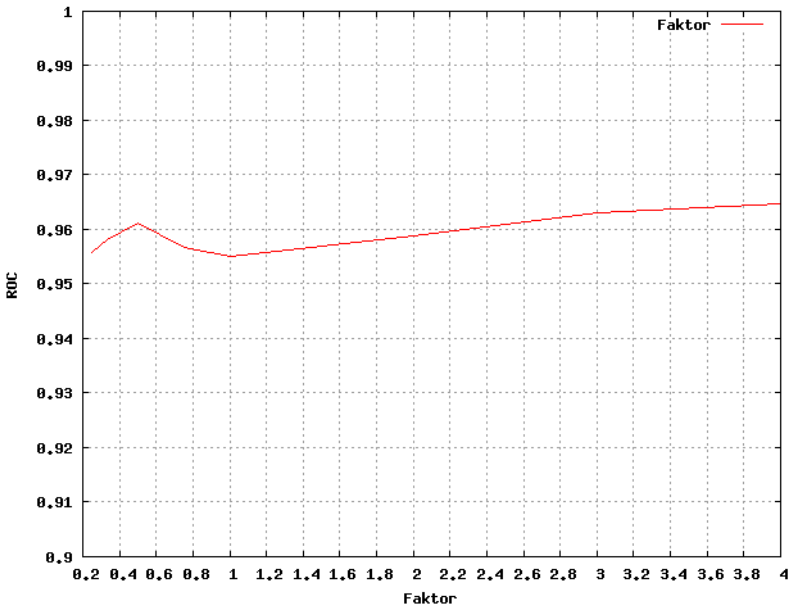

Abbildung 6.10: Combined Parameter ROC Datensatz 2

6.2.3 Datensatz 3

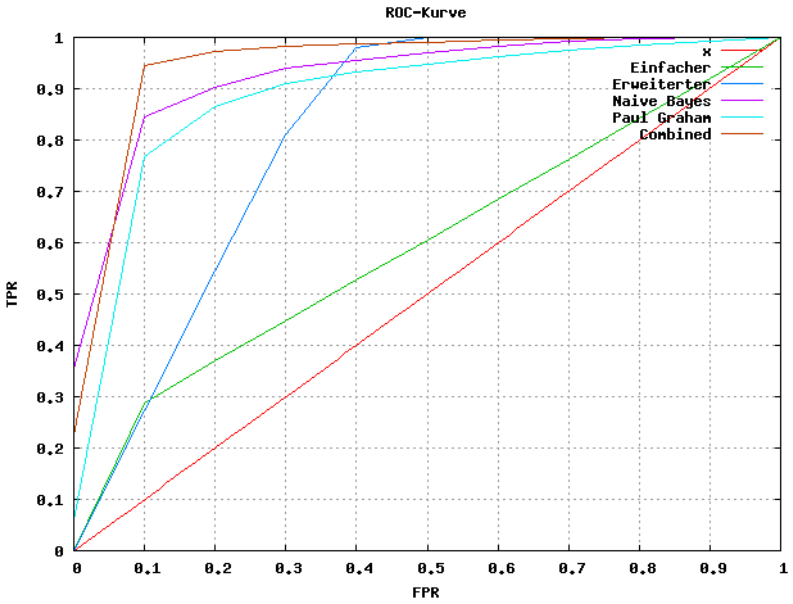

Abbildung 6.11: ROC-Kurven Datensatz 3

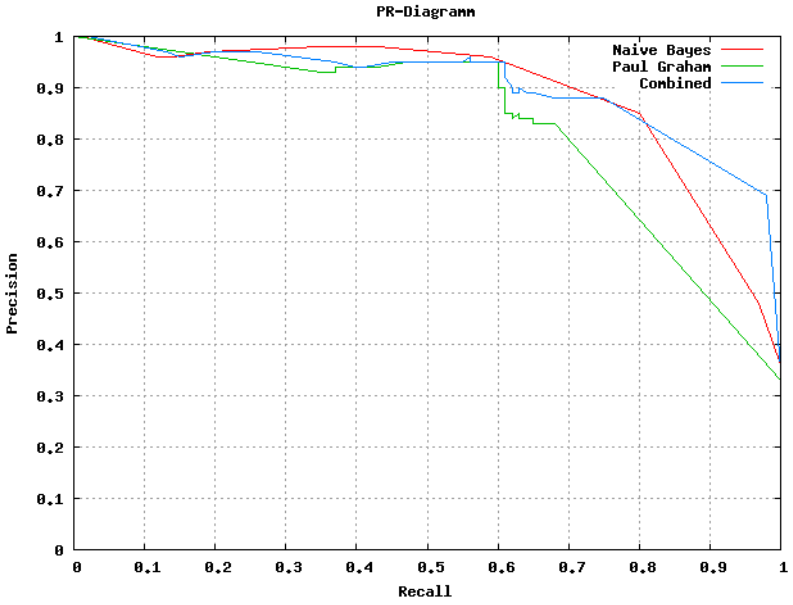

Abbildung 6.12: PR-Diagramm Datensatz 3

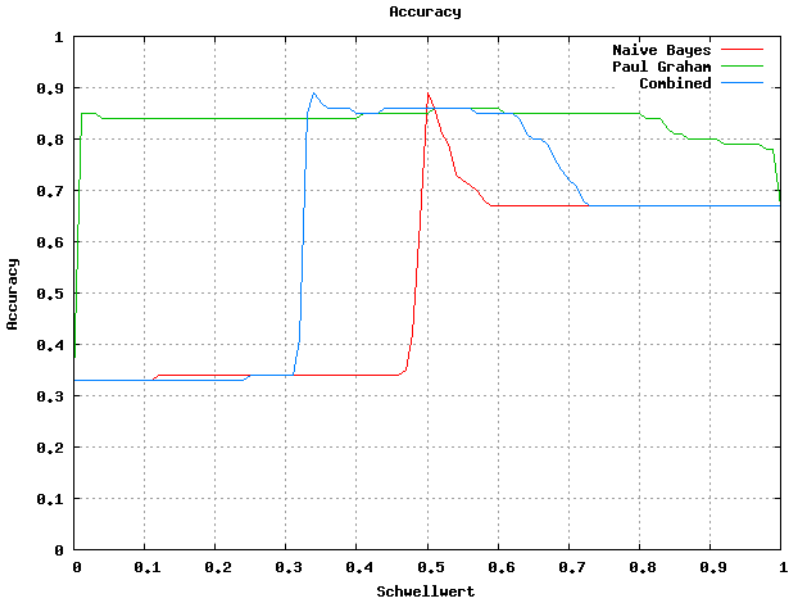

Abbildung 6.13: Accuracy-Diagramm Datensatz 3

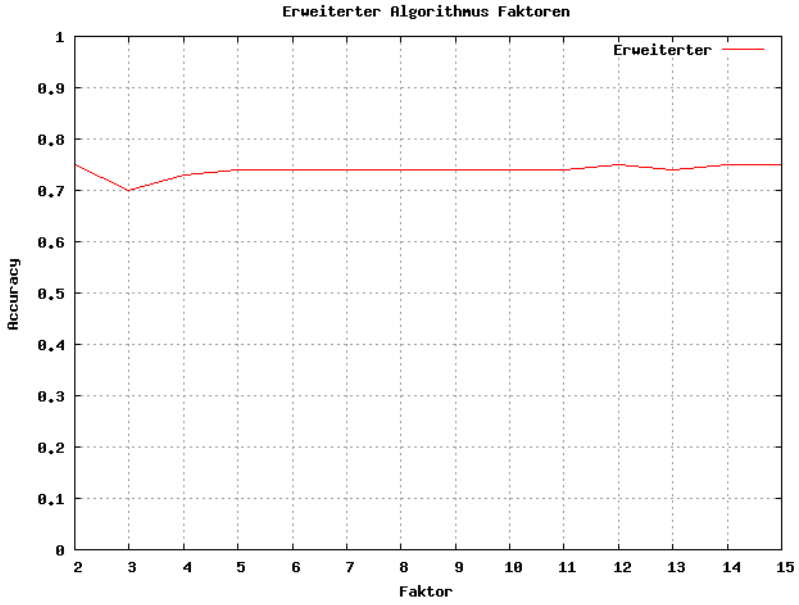

Abbildung 6.14: Erweiterter Algorithmus-Diagramm Datensatz 3

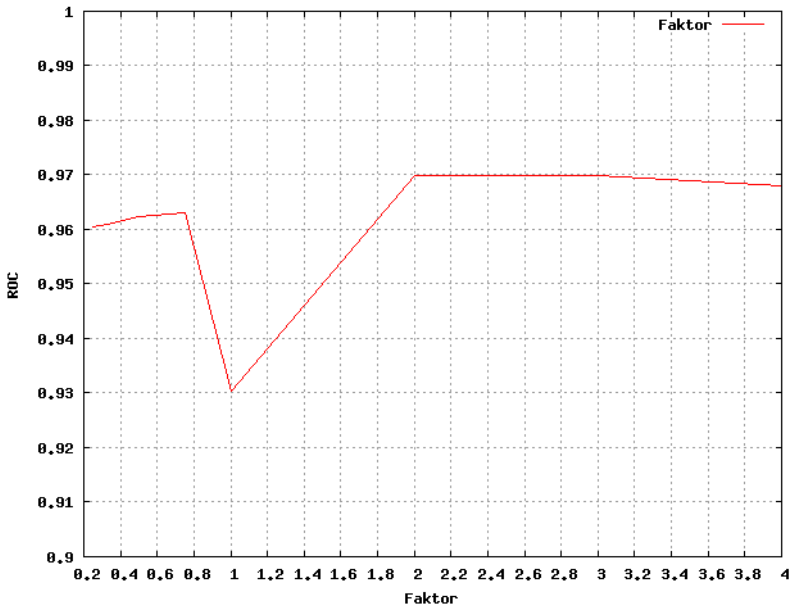

Abbildung 6.15: Combined Parameter ROC Datensatz 3

6.2.4 Datensatz 4

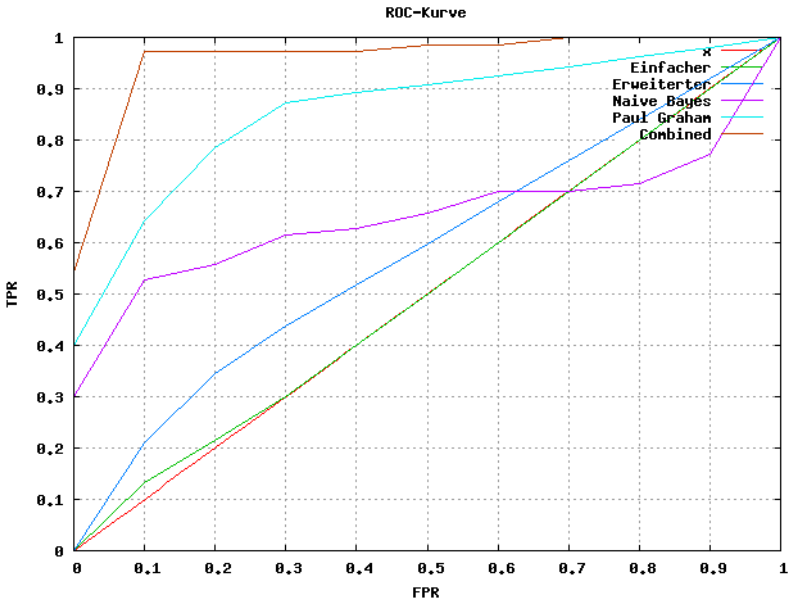

Abbildung 6.16: ROC-Kurven Datensatz 4

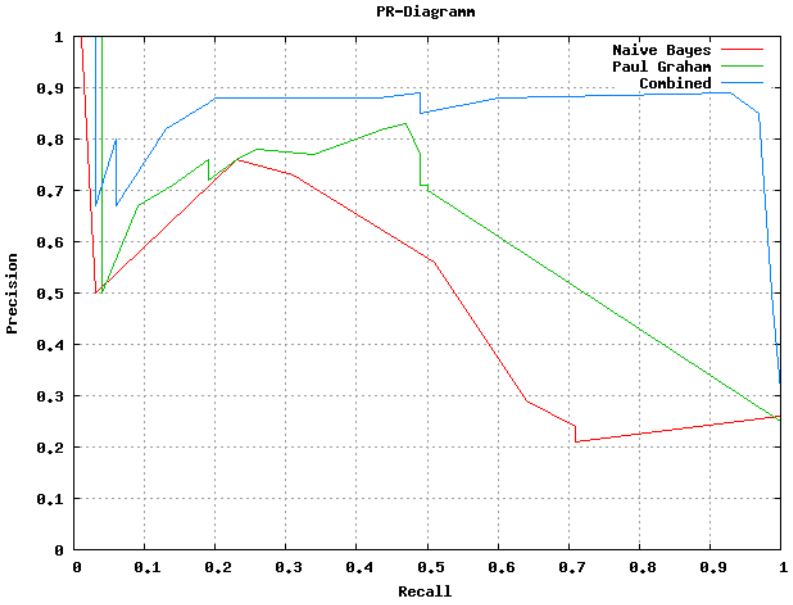

Abbildung 6.17: PR-Diagramm Datensatz 4

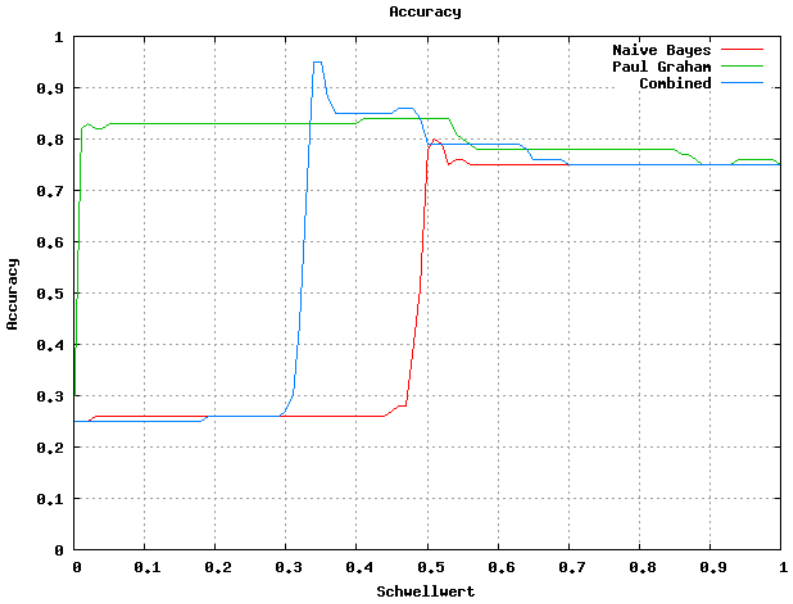

Abbildung 6.18: Accuracy-Diagramm Datensatz 4

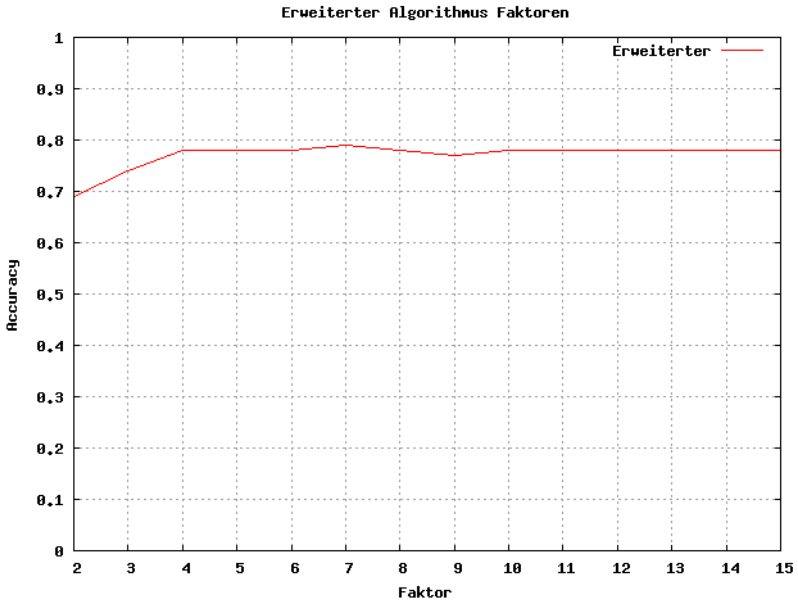

Abbildung 6.19: Erweiterter Algorithmus-Diagramm Datensatz 4

6.2.5 Datensatz 5

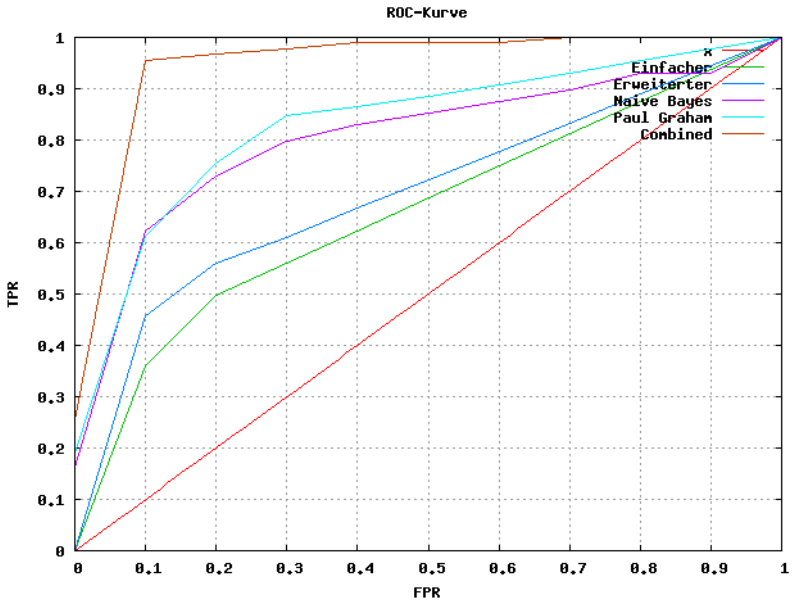

Abbildung 6.20: ROC-Kurven Datensatz 5

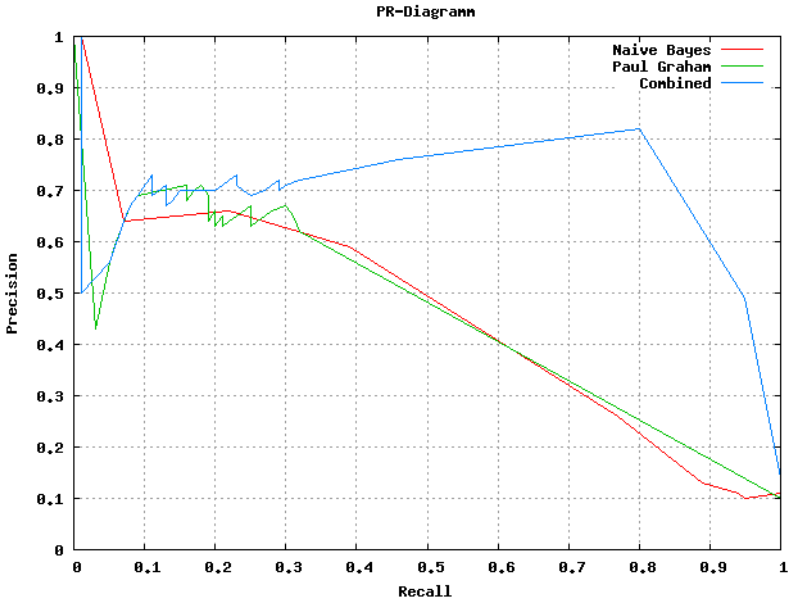

Abbildung 6.21: PR-Diagramm Datensatz 5

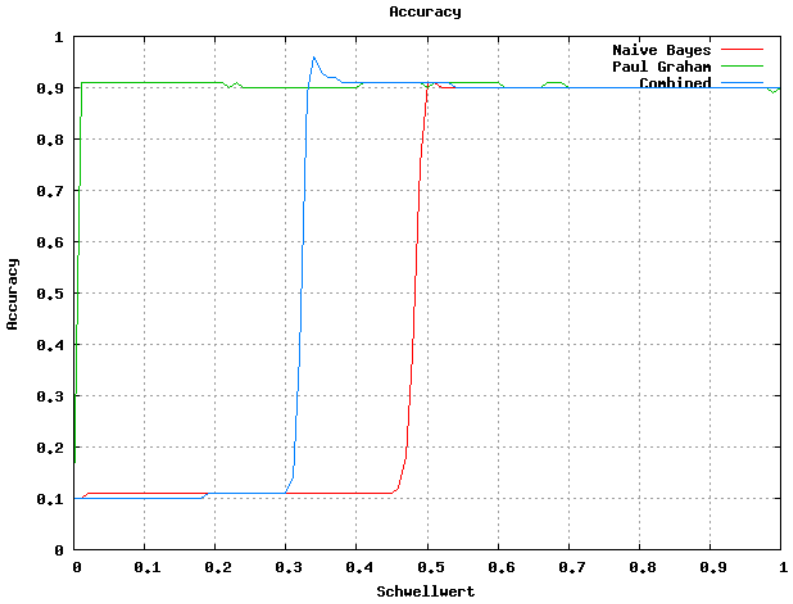

Abbildung 6.22: Accuracy-Diagramm Datensatz 5

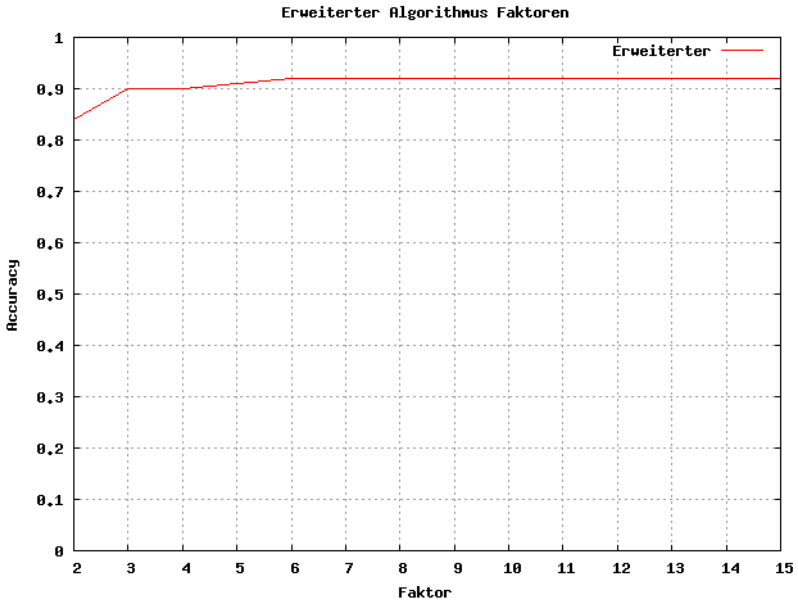

Abbildung 6.23: Erweiterter Algorithmus-Diagramm Datensatz 5

Algorithmenverzeichnis

Tabellenverzeichnis

Abbildungsverzeichnis

Literaturverzeichnis

[auc] *The Area Under an ROC Curve*. http://gim.unmc.edu/dxtests/roc3.htm. [Online; zugegriffen am 24.11.2008].

[CDM08] Christopher D. Manning, Prabhakar Raghavan, Hinrich Schütze: *Introduction to Information Retrieval*. Cambridge University Press, 2008.

[Cha03] Chakrabi, Soumen: *Mining the Web*. Morgan Kaufmann Publishers, 2003.

[Coh05] Cohen, William W.: *Enron Email Dataset*. http://www.cs.cmu.edu/~enron/, 2005. [Online; zugegriffen am 26.01.2009].

[con08] *Konfusionsmatrix*. http://de.wikipedia.org/wiki/Konfusionsmatrix, 2008. [Online; zugegriffen am 20.12.2008].

[DJH01] David J. Hand, Robert J. Till: *A Simple Generalisation of the Area Under the ROC Curve for Multiple Class Classification Problems*, 2001.

[fin] *Python Documentation RE*. http://docs.python.org/dev/library/re.html. [Online; zugegriffen am 01.01.2009].

[Ghi] Ghiglieri, Marco. http://www.mgnis.de/ma/ma.htm. [Online; zugegriffen am 04.01.2009].

[gma07] *Gmail attachment reminder*. http://userscripts.org/scripts/show/2419, 2007. [Online; zugegriffen am 24.11.2008].

[GNU] GNUPlot. http://www.gnuplot.info/. [Online; zugegriffen am 04.01.2009].

[Gra02] Graham, Paul: *A plan for spam*. http://www.paulgraham.com/spam.html, 2002. [Online; zugegriffen am 19.10.2008].

[Gra03] Graham, Paul: *Better Bayesian Filtering*. http://www.paulgraham.com/better.html, 2003. [Online; zugegriffen am 05.01.2009].

[H.O08] H.Ogi: *Check and Send*. https://addons.mozilla.org/de/thunderbird/addon/2281, 2008. [Online; zugegriffen am 24.11.2008].

[Mit97] Mitchell, Tom M.: *Machine Learning*. McGRAW-HILL, 1997.

[Moz] Mozilla: *Thunderbird-Add-ons*. https://addons.mozilla.org/de/thunderbird. [Online; zugegriffen am 24.11.2008].

[Moz07] Mozilla: *The Joy of XUL*. https://developer.mozilla.org/en/The_Joy_of_XUL, 2007. [Online; zugegriffen am 26.01.2009].

[Moz08a] Mozilla: *Developer Center Building an Extension*. https://developer.mozilla.org/En/Building_an_Extension, 2008. [Online; zugegriffen am 02.01.2009].

[Moz08b] Mozilla: *Mozilla Developer Center*. http://developer.mozilla.org/en/Extensions, 2008. [Online; zugegriffen am 19.10.2008].

[Moz08c] Mozilla: *Mozilla Thunderbird*. http://www.mozilla-europe.org/de/products/thunderbird/, 2008. [Online; zugegriffen am 24.11.2008].

[PK08] Philipp Kewisch, Daniel Folkinshteyn: *Attachment Reminder*. https://addons.mozilla.org/de/thunderbird/addon/5759, 2008. [Online; zugegriffen am 24.11.2008].

[pyx] *PyXPCOMext*. http://pyxpcomext.mozdev.org/. [Online; zugegriffen am 01.01.2009].

[rfc] *RFC 2822*. http://www.ietf.org/rfc/rfc2822.txt. [Online; zugegriffen am 01.01.2009].

[Tsc08] Tschabitscher, Heinz: *About.com*. http://email.about.com/od/emailtrivia/f/emails_per_day.htm, 2008. [Online; zugegriffen am 04.11.2008].

[Wik07] Wikimedia: *Cross Platform Component Object Model*. http://de.wikipedia.org/wiki/XPCOM, 2007. [Online; zugegriffen am 19.10.2008].