Федеральное государственное автономное
образовательное учреждение высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт математики и фундаментальной информатики
Базовая кафедра вычислительных и информационных технологий

УТВЕРЖДАЮ
Заведующий кафедрой
_____ / В.В. Шайдуров

«16» июня 2016 г.

БАКАЛАВРСКАЯ РАБОТА

Направление 02.03.01 Математика и компьютерные науки

ИССЛЕДОВАНИЕ АЛГОРИТМОВ РЕШЕНИЯ ЗАДАЧИ ФИЛЬТРАЦИИ СПАМА

Научный руководитель
кандидат физико-математических наук,
доцент _____ / И.В. Баранова
16.06.2016

Выпускник _____ / А.В. Кулистов
16.06.2016

Красноярск 2016

СОДЕРЖАНИЕ

ВВЕДЕНИЕ

Одним из самых востребованных направлений в области защиты информации является разработка методов фильтрации потока электронной почты. В настоящее время электронная почта стала одним из наиболее распространенных средств связи, управления и бизнеса. Она достаточно совершенна в техническом отношении и представляет собой недорогую альтернативу привычным средствам связи.

Вместе с развитием электронной почты увеличивается и количество угроз ее нормальному функционированию. Наиболее серьезной и важной проблемой стал так называемый спам.

Спам (англ. *spam*) − массово распространяемые рекламные материалы или подобные коммерческие виды сообщений лицам, не выражавшим желания их получать.

Целью работы является исследование и реализация основных алгоритмов решения задачи фильтрации спама.

В работе приводится постановка задачи фильтрации спама, которая является частным случаем задачи классификации данных. Рассматриваются основные подходы к решению задачи фильтрации.

Особое внимание уделяется одному из наиболее востребованных методов классификации – наивному байесовскому классификатору. Данный метод выполняет анализ текстовых данных, что позволяет использовать его для нахождения решения поставленной задачи фильтрации спама. В работе рассматриваются две разновидности байесовского классификатора: на основе многомерной и мультиномиальной модели.

В работе предлагается новый метод фильтрации данных на основе модели «мешка слов».

Проводится сравнение вышеперечисленных методов по вычислительной сложности и точности нахождения решения.

В работе решается практическая задача фильтрации спама – выполняется классификация электронных писем на основании их содержимого. Данная задача решается с помощью байесовского классификатора и предложенного метода. Проводится визуализация результатов работы методов, и анализируются полученные итоги.

Разработано программное приложение, реализующее работу трёх вышеперечисленных алгоритмов фильтрации спама.

1. Задача фильтрации спама

В данной работе будет решаться задача фильтрации спама, распространяемого по электронной почте. Как уже было сказано ранее, спам представляет собой нежелательные массовые рассылки сообщений, в основном рекламного характера.

Определение 1.1 Спам (англ. *spam*) − массово распространяемые рекламные материалы или подобные коммерческие виды сообщений лицам, не выражавшим желания их получать.

В общепринятом значении термин в русском языке впервые стал употребляться применительно к рассылке электронных писем. Доля спама в мировом почтовом трафике составляет от 60% до 80%.

Среди стран-источников спама лидируют США (16,7%), за которыми с существенным отрывом следует Россия (6%).

1.1 Основные подходы к решению задачи фильтрации спама

На сегодняшний день разработан ряд технологий построения фильтров спама – специальных сервисов для отсеивания нежелательной корреспонденции. Все технологии можно разделить на настраиваемые вручную и интеллектуальные (автоматические). Настраиваемые вручную фильтры основываются на списках доступа и настраиваются непосредственно пользователем, который задаёт либо нежелательные адреса, при политике пропуска по «черному списку», либо разрешенные адреса, при политике пропуска по «белому списку». Однако ручные способы фильтрации нежелательных сообщений малоэффективны и требуют постоянного обновления списков доступа, создавая дополнительную нагрузку на пользователя.

Решения, которые в той или иной мере могут помочь снять проблему спама, можно условно разделить на следующие группы:

- Простейшие способы ручной или автоматической фильтрации почты по заголовкам.

Любой пользователь может перейти с протокола POP3 на IMAP4 или на Web-интерфейс и оценивать электронные письма только по их заголовкам, не получая текста. Во многих почтовых программах можно настроить автоматическую фильтрацию спама по заголовкам писем. Однако в последнем случае требуется очень тонкая и трудоёмкая подгонка условий оценки, а также внимательность к оформлению сообщений от ваших постоянных корреспондентов.

- Специальные службы фильтрации спама, которые могут находиться у почтового провайдера или на отдельном сервере (последние, как правило, платные).

Этот способ является более надежным, чем предыдущий. В некоторых случаях вся почта отправляется на определенный адрес, где фильтруется, и к пользователю приходит уже очищенной. Данный метод с точки зрения трудозатрат пользователя является самый простым, но, как правило, и наименее контролируемым (велика вероятность, что может потеряться часть полезной корреспонденции, о чем никто никогда не узнает).

- Входные антиспам-фильтры, основанные на анализе IP-адреса хоста, передающего спам (который можно узнать, например, по отзывам пострадавших), и использование общих баз данных с адресами таких спамеров (DNSBL — DNS Black Lists).

Однако в данный момент этот способ борьбы с современными методами спама является малоэффективным, вследствие частого обновления адресов спамеров.

- Фильтрация на основе автоматического пополнения access-листа адресами спамеров.

- Программы или встраиваемые модули для анализа содержимого письма.

Программы для такой проверки (их может быть несколько) принимают информацию от почтовой программы, а возвращают, как правило, свою оценку и рекомендацию к дальнейшему действию.

Целью данной работы является изучение и реализация алгоритмов фильтрации спама, относящихся к последней группе. В работе рассматриваются следующие методы фильтрации спама: наивный байесовский классификатор и классификация на основе модели «мешка слов».

1.2 Постановка задачи

Фильтрация спама является разновидностью задачи классификации текстов, поэтому будем использовать следующую модель задачи классификации.

Постановка задачи классификации текстов:

Пусть имеется $D = \{d_1, d_2, \dots, d_m\}$ − множество текстовых документов с набором признаков $W = \{w_1, w_2, \dots, w_n\}$ (т.е. каждый текстовый документ $d_i = \left(w_i^1, w_i^2, \dots, w_i^n\right)$ и задана функция расстояния (метрика) между объектами $\rho\left(d_i, d_j\right), d_i, d_j \in D$.

Задано множество классов $C = \{C_1, C_2, \dots, C_k\}, k \leq m,$ которое представляет собой разбиение множества объектов такое, что класс $C_i \cap C_j = \emptyset$ $i \neq j$. В нашей задаче $k = 2$.

Определение 1.2 ***Функцией классификации*** называется функция $f: D \to Y,$ которая каждому объекту $d \in D$ ставит в однозначное соответствие номер кластера $y \in Y = \{1, \dots, k\}, k \leq m.$

Класс $C_i = \{d \in D, f(d) = i\}.$

Задано конечное множество объектов $\widetilde{D} \subset D$, для которых известно, к каким классам они относятся. Это множество называется ***обучающей выборкой***. Классовая принадлежность остальных объектов не известна.

Тогда постановку задачу классификации данных можно сформулировать следующим образом:

Требуется найти такую функцию f^*, чтобы

$$Q(f^*, C, \rho) = \min_f Q(f, C, \rho),$$

где $Q(f, C, \rho)$ – выбранный критерий качества классификации.

Наиболее часто в задачах классификации или кластеризации текстовых данных используется байесовский классификатор.

Во всех приведенных ниже методах перед проведением кластеризации проводится предварительная обработка текстов – из них удаляются *стоп-слова*, как шумовые составляющие.

Стоп-слова − слова, знаки или символы, которые не несут самостоятельной смысловой нагрузки и просто игнорируются поисковыми системами при осуществлении ранжирования или индексации сайтов. К ним относят союзы и союзные слова, местоимения, предлоги, частицы, междометия, указательные слова, цифры, знаки препинания, вводные слова, ряд некоторых существительных, глаголов и наречий.

2. Наивный байесовский классификатор

2.1 Характеристика метода

Среди программ, предназначенных для борьбы со спамом, особенно интересны те, что работают по принципам Байеса и самообучаются в процессе анализа корреспонденции.

Байесовский классификатор — широкий класс алгоритмов классификации, основанный на принципе максимума апостериорной вероятности. Для классифицируемого объекта вычисляются ***функции правдоподобия*** каждого из классов, по ним вычисляются апостериорные

вероятности классов. Объект относится к тому классу, для которого апостериорная вероятность максимальна.

Данная технология отличается использованием байесовских принципов для распознавания спама по образу, моделирование которого происходит благодаря анализу самого спама. Однако простота применения байесовских принципов обманчива, так как отнесение письма к спаму производится по сложным алгоритмам выявления общих элементов в реальных посланиях. Таким образом, чем большее количество спама подверглось анализу, тем лучше работает фильтр. Кроме того, метод Байеса обладает автокоррекцией, поскольку в случае изменения структуры писем фильтр изменяется автоматически.

При обучении антиспам-фильтра по методу Байеса для каждого встреченного в письмах слова высчитывается и сохраняется его «вес» — вероятность того, что письмо с этим словом является спамом.

Отнесение письма к «спаму» или к обычной корреспонденции производится по тому, превышает ли его «вес» некую планку, заданную пользователем (обычно берут 60-80%). После принятия решения по письму в базе данных обновляются «веса» для вошедших в него слов.

Алгоритмы данного метода фильтрации спама элементарны, удобны, достаточно эффективны (при условии обучения на достаточно большом количестве писем блокирует до 95-97% спама) и обучаемы. На основе данного метода функционирует большинство современных спам-фильтров, установленных как на почтовых серверах, так и встроенных в почтовое программное обеспечение пользователя.

Однако у метода есть и принципиальные недостатки: во-первых, он базируется на предположении, что одни слова чаще встречаются в спаме, а другие — в обычных письмах, и неэффективен, если данное предположение неверно; во-вторых, данный метод фильтрации спама работает только с текстом, что позволяет спамерам обходить его, включая рекламную

информацию не в тело письма, а в графическое вложение, сопровождая само письмо либо бессмысленным, либо нейтральным текстом.

2.2 Описание принципа работы наивного байесовского классификатора

Данная модель классификации базируется на понятии условной вероятности принадлежности документа d классу C_j.

В основе байесовского классификатора лежит теорема Байеса.

Теорема Байеса: одна из основных теорем теории вероятностей, которая позволяет определить вероятность какого-либо события A при условии, что произошло другое статистически взаимозависимое с ним событие B:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)},$$

т.е. она позволяет вычислить условную вероятность $P(A|B)$ того, что имело место событие A, если в результате эксперимента наблюдалось событие B, при известных вероятностях наступления событий – $P(A)$ и $P(B)$, и условной вероятности наступления события A при существующем B – $P(B|A)$.

Пусть для каждого класса $C_j \in C$ известна *априорная вероятность* $P(C_j)$ того, что появится объект класса C_j, и плотности распределения $P(d|C_j)$ каждого из классов, называемые также *функциями правдоподобия* классов.

Согласно теореме Байеса, вероятность принадлежности документа d к классу C_j вычисляется следующим образом:

$$P(C_j|d) = \frac{P(d|C_j)P(C_j)}{P(d)},$$

где $P(C_j|d)$ – вероятность, что документ d принадлежит классу C_j, именно ее нам необходимо вычислить; $P(d|C_j)$ – вероятность встретить документ d среди документов класса C_j *(плотность распределения* класса C_j*)*;

$P(C_j)$ – априорная (безусловная) вероятность класса появления документа класса C_j; $P(d)$ – безусловная вероятность появления документа d в корпусе документов.

Значение $P(C_j|d) = P(d|C_j)P(C_j)$ интерпретируется как апостериорная вероятность того, что объект d принадлежит классу $C_j \in C$.

Цель классификации состоит в том, чтобы понять к какому классу принадлежит документ.

Требуется найти такую функцию классификации f^*, чтобы

$$Q(f^*, C, \rho) = \min_f Q(f, C, \rho),$$

где $Q(f, C, \rho)$ – выбранный критерий качества классификации.

Наиболее вероятным классом C^*, к которому принадлежит документ d, является тот класс, для которого условная вероятность принадлежности документа d классу C_j максимальна:

$$C^* = \arg\max_j P(C_j|d).$$

Необходимо вычислить вероятность для всех классов и выбрать тот класс, для которого вероятность имеет максимальное значение.

По теореме Байеса:

$$C^* = \arg\max_j \frac{P(d|C_j)P(C_j)}{P(d)}.$$

Согласно решаемой нами задачи классификации каждый документ $d \in D = \{d_1, d_2, \dots, d_m\}$ задан признаками из $W = \{w_1, w_2, \dots, w_n\}$, т.е. каждый текстовый документ $d_i = (w_i^1, w_i^2, \dots, w_i^n)$. Для данной модели признаками документа мы будем считать некоторые характеристики, связанные со словами, содержащимися в нем. Чаще всего считается, что w_i – вес i-ого термина, а n – размер словаря выборки. В следующем разделе рассмотрим подробнее способы задания признаков w_i для каждого документа.

Так как $d = \{w_1, w_2, \dots, w_n\}$, то

$$C^* = \arg\max_j \frac{P(w_1, w_2, \dots, w_n|C_j)P(C_j)}{P(d)}.$$

Далее делается существенное допущение, которое и объясняет, почему этот алгоритм называют наивным. Оно звучит следующим образом: знаменатель может быть опущен, так как для одного и того же документа d вероятность $P(d)$ будет одинаковой, а это значит, что ее можно не учитывать:

$$C^* = \arg\max_j P(w_1, w_2, \dots, w_n|C_j)P(C_j).$$

Также в модели наивного байесовского классификатора предполагается, что все признаки $w_1, w_2, \dots, w_n$ документа d независимы друг от друга. При этом вносится уточнение, что позиция термина в предложении не важна.

Таким образом, условную вероятность $P(w_1, w_2, \dots, w_n|C_j)$ для признаков $w_1, w_2, \dots, w_n$ можно представить в следующем виде:

$$P(w_1, w_2, \dots, w_n|C_j) = \prod_i P(w_i|C_j).$$

Таким образом, для нахождения наиболее вероятного класса для документа $d = \{w_1, w_2, \dots, w_n\}$ с помощью наивного байесовского классификатора, необходимо вычислить условные вероятности принадлежности документа d для каждого из представленных классов и выбрать класс, имеющий максимальную вероятность (**принцип максимума апостериорной вероятности**):

$$C^* = \arg\max_j \left[P(C_j) \prod_i P(w_i|C_j) \right], j = 1,2.$$

Вероятность каждого класса можно оценить частотой встречаемости документов этого класса в обучающей выборке: $P(C_j)$ оценивается отношением количества документов класса j в обучающей выборке к общему количеству документов в выборке:

$$P(C_j) = \frac{D_{C_j}}{D},$$

где D_{C_j} – количество документов класса C_j, а D – общее количество документов в выборке.

Для оценки *плотность распределения* классов - условных вероятностей встречаемости слов $P(w_i|C_j)$ используются два разных подхода, которые мы рассмотрим в следующих разделах.

Преимущества наивного байесовского классификатора:

- Обработка количественных и дискретных данных.
- Устойчивые и изолированные точки шума.
- Обработка отсутствующих значений путем их игнорирования.
- Прогноз подсчетов во время вычисления вероятности.
- Быстрота и эффективность относительно пространства.
- Отсутствие чувствительности к посторонним функциям.
- Квадратическая граница решений.

Недостатки метода:

- Предполагает независимость функций

Приведем последовательность действий при решении задачи алгоритмом Байеса:

- Сначала необходимо удалить или заменить все известные и неизвестные специальные символы (знаки препинания, абзацы т.д.), тем самым только слова могут быть использованы для подсчета вероятности.
- Необходимо выделить каждое слово в тексте w_iи определить вероятность отнесения слова к спаму $P(w_i/C_1)$
- Аналогичным образом определяем вероятность отнесения слова к не спаму $P(w_i/C_2)$
- Данные вероятности оцениваются частотой встречаемости слова w_i соответственно в спам письмах и не спам письмах, которые получены в результате работы алгоритма на обучающей выборке.
- Получить финальную вероятность по формуле:

$$C^* = \arg\max_j \left[P(C_j) \prod_i P(w_i|C_j) \right], j = 1,2.$$

- Документ будет отнесен к классу с максимальной вероятностью.
- После отнесения документа к соответствующему классу происходит изменение частот встречаемости слов в спам и не спам письмах. Если значение не присутствовало в базе данных, происходит его добавление.

2.3 Многомерная модель наивного байесовского классификатора

Как уже было сказано ранее, в нашей задаче каждый текстовый документ характеризуется некоторыми признаками $d_i = (w_i^1, w_i^2, \dots, w_i^n)$. Признаками документа мы будем считать характеристики, связанные со словами, содержащимися в документе.

В многомерной модели вначале фиксируется словарь терминов, встречающихся во всех документах обучающей выборки. Затем для каждого документа определяется, встретилось ли в документе то или иное слово. При этом количество повторений каждого слова не учитывается.

Таким образом, в данной модели каждому документу ставится в соответствие вектор бинарных атрибутов, показывающих, встретилось ли в документе то или иное слово.

Когда мы подсчитываем функцию правдоподобия принадлежности документа d_i классу C_j, мы перемножаем вероятности того, что встретилось каждое слово из документа и вероятности того, что не встретилось каждое (словарное) слово, которое не встретилось. Получается модель многомерных испытаний Бернулли. Наивное предположение в том, что события «встретилось ли слово» предполагаются независимыми.

Для применения требуется зафиксировать словарь, а количество повторений каждого слова теряется.

Страница изъята

Страница изъята

Страница изъята

Страница изъята

Страница изъята

$$\rho(d_i, d_j) = \sqrt{\sum_{k=1}^{n} \left(d_i^k - d_j^k\right)^2}.$$

Если два текста близки друг другу, то расстояние между соответствующими им векторами будет наименьшим.

3. Практический пример задачи фильтрации спама

Имелась обучающая выборка писем, состоящая из обычных и спам-писем. Размер обучающей выборки составил 100 писем, из которых половина являлась спамом.

Практическая задача заключалась в том, чтобы проверить все три программы, написанные на основе данных трех методов, на точность решения задачи фильтрации спама. Каждой программе на вход подавалось по 150 писем, часть из которых была спамом, часть не спамом и еще одна часть – письма, которые являлись не спамом, но содержали слова, часто встречающиеся в спаме. Все результаты фиксировались экспертом и после составлялась сравнительная таблица по работе данных методов.

В работе было создано программное обеспечение, реализующее работу алгоритмов: многомерного наивного байесовского классификатора, мультиномиального наивного байесовского классификатора и метода, основанного на модели «мешка слов».

3.1 Результаты решения практической задачи

В таблицах 1-3 приведены результаты работы данных алгоритмов для 3 тестовых писем: первое письмо являлось спамом, второе – нет, третье письмо не являлось спамом, хотя и содержало некоторые слова, часто встречающиеся в спаме.

Таблица 1 – Письмо №1 (не спам)

	Вероятность отнесения к спаму	Вероятность отнесения к не спаму	Результат
Многомерный наивный байесовский классификатор	0,4575	0,5425	Не спам
Мультиномиальный наивный байесовский классификатор	0,4987	0,5013	Не спам
Классификация на основе модели «мешка слов»			Не спам

Таблица 2 – Письмо №2 (спам)

	Вероятность отнесения к спаму	Вероятность отнесения к не спаму	Результат
Многомерный наивный байесовский классификатор	0,503	0,497	Спам
Мультиномиальный наивный байесовский классификатор	0,518	0,4819	Спам
Классификация на основе модели «мешка слов»			Спам

Таблица 3 – Письмо №3 (неопределенное письмо)

	Вероятность отнесения к спаму	Вероятность отнесения к не спаму	Результат
Многомерный наивный байесовский классификатор	0,4661	0,5339	Не спам
Мультиномиальный наивный байесовский классификатор	0,5019	0,4981	Спам
Классификация на основе модели «мешка слов»			Не спам

В таблице 4 приведены результаты работы алгоритмов для данного множества писем.

Таблица 4 – Результаты работы алгоритмов для данного множества писем.

	Количество не спам писем	Количество спам писем	Количество верно отнесенных писем
Многомерный наивный байесовский классификатор	100	50	120
Мультиномиальный наивный байесовский классификатор	100	50	119
Классификация на основе модели «мешка слов»	100	50	99

3.2 Результаты работы программы, решающей практическую задачу многомерным наивным байесовским классификатором на тестовых письмах

Результаты работы программы, основанной на многомерном наивном байесовском классификаторе для 3 тестовых писем (первое письмо являлось не спамом, второе было спамом, третье письмо не являлось спамом, хотя и содержало некоторые слова, часто встречающиеся в спаме), представлены на Рисунках 1-3. Из этих результатов видно, что письма, поданные на вход, были отнесены к соответствующим классам верно.

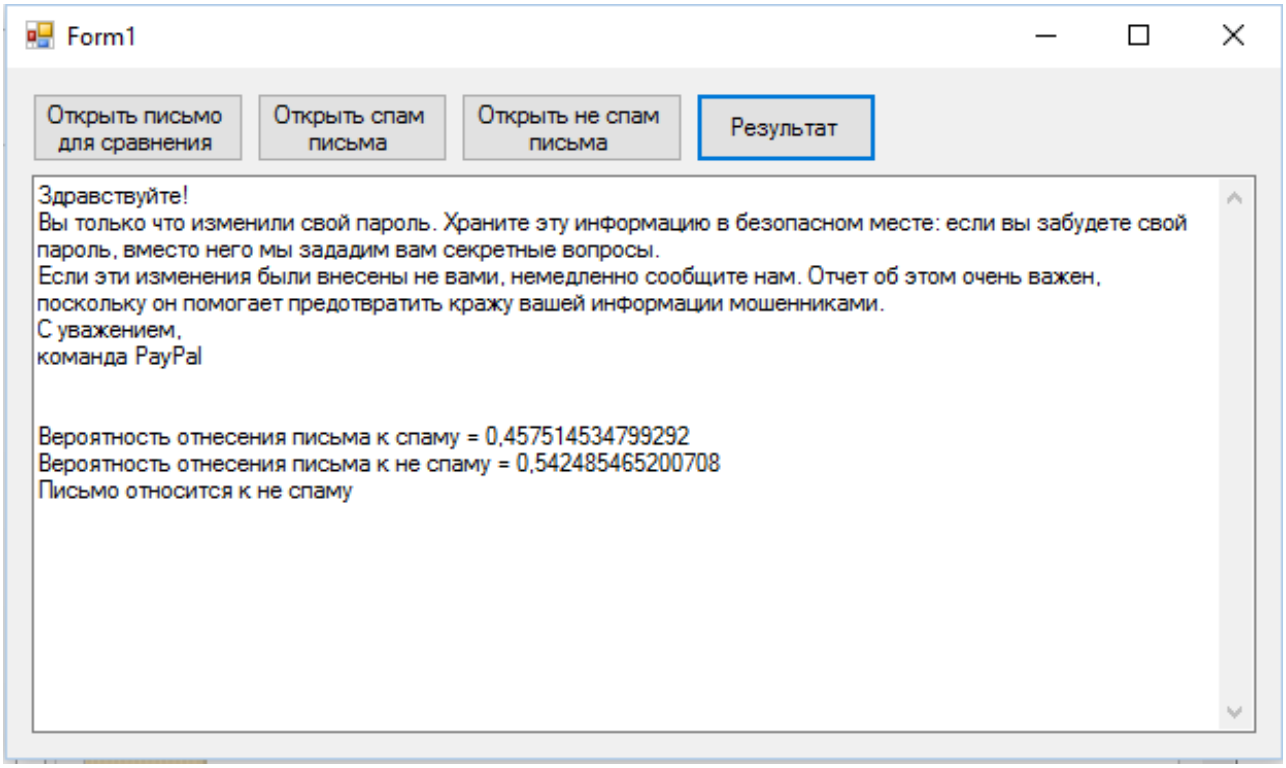

Рисунок 1 – Результат работы многомерного наивного байесовского классификатора для письма, которое относится к не спаму

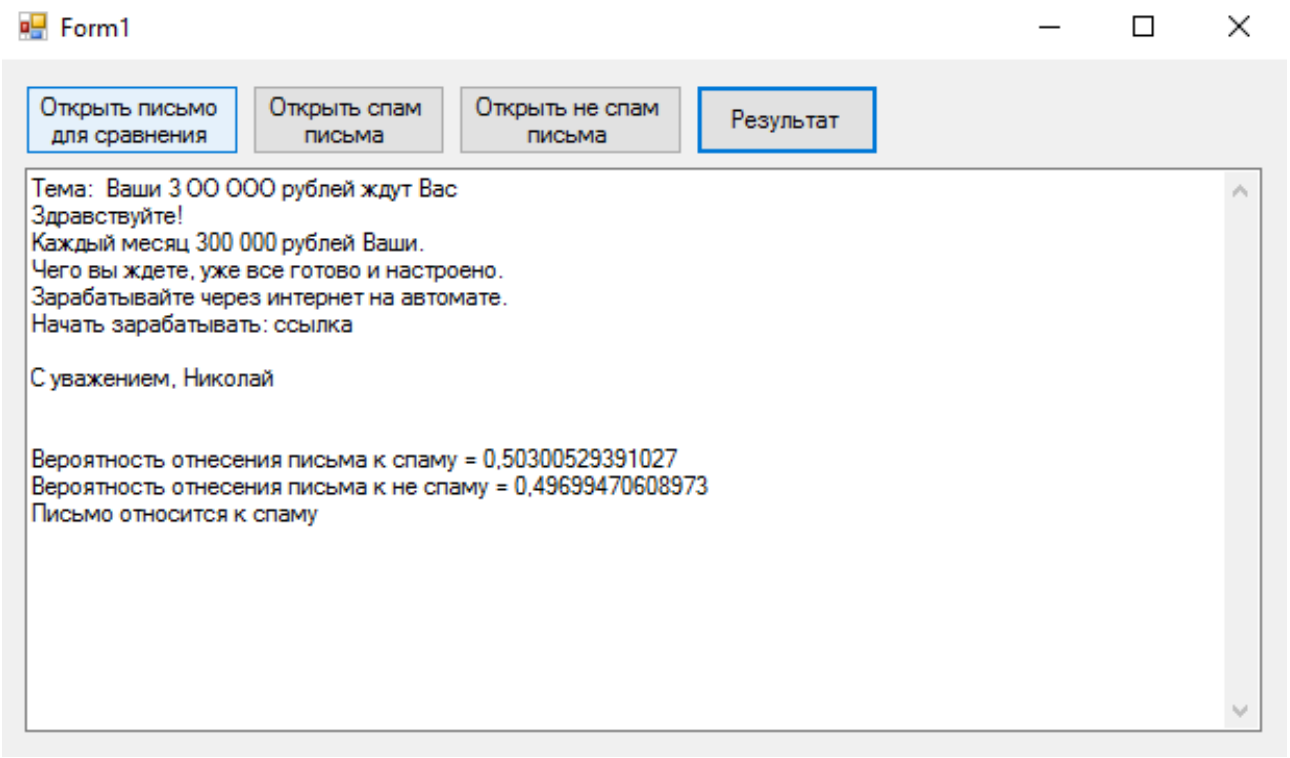

Рисунок 2 – Результат работы многомерного наивного байесовского классификатора для письма, которое относится к спаму

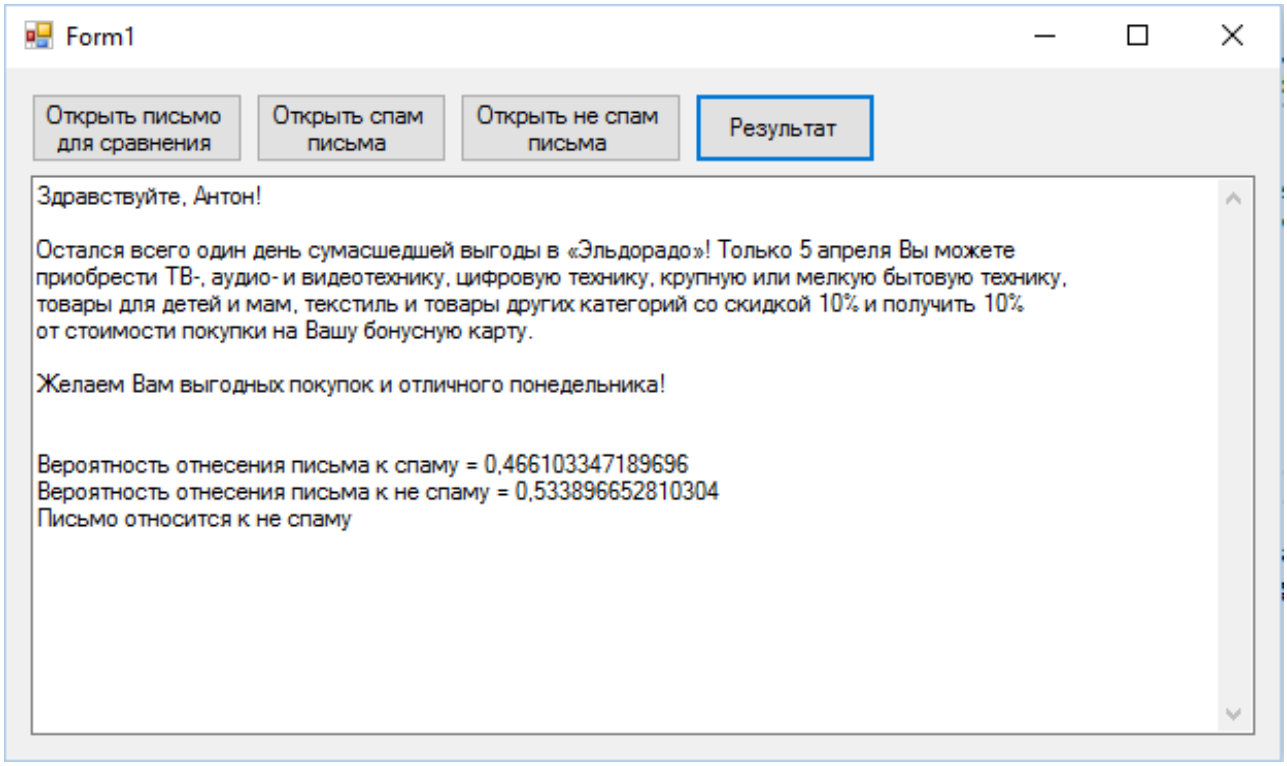

Рисунок 3 – Результат работы многомерного наивного байесовского классификатора для письма, которое является не спамом, хотя и содержит некоторые слова, часто встречающиеся в спаме

3.3 Результаты работы программы, решающей практическую задачу мультиномиальным наивным байесовским классификатором на тестовых письмах

Результаты работы программы, основанной на мультиномиальном наивном байесовском классификаторе для 3 тестовых писем (первое письмо являлось не спамом, второе было спамом, третье письмо не являлось спамом, хотя и содержало некоторые слова, часто встречающиеся в спаме), представлены на Рисунках 4-6. Из результатов работы данного метода видно, что два письма из трех были отнесены к классам верно, но письмо, которое является не спамом, хотя и содержит некоторые слова, часто встречающиеся в спаме было отнесено к спаму.

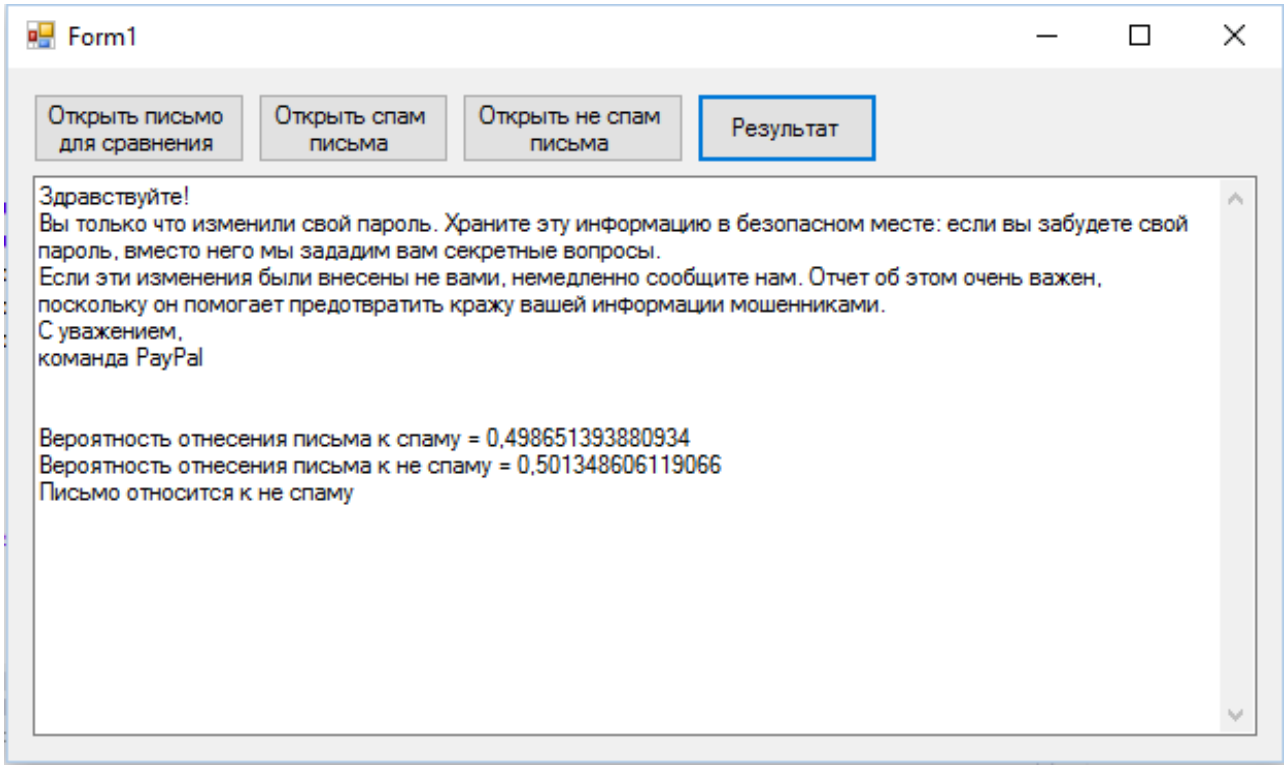

Рисунок 4 – Результат работы мультиномиального наивного байесовского классификатора для письма, которое относится к не спаму

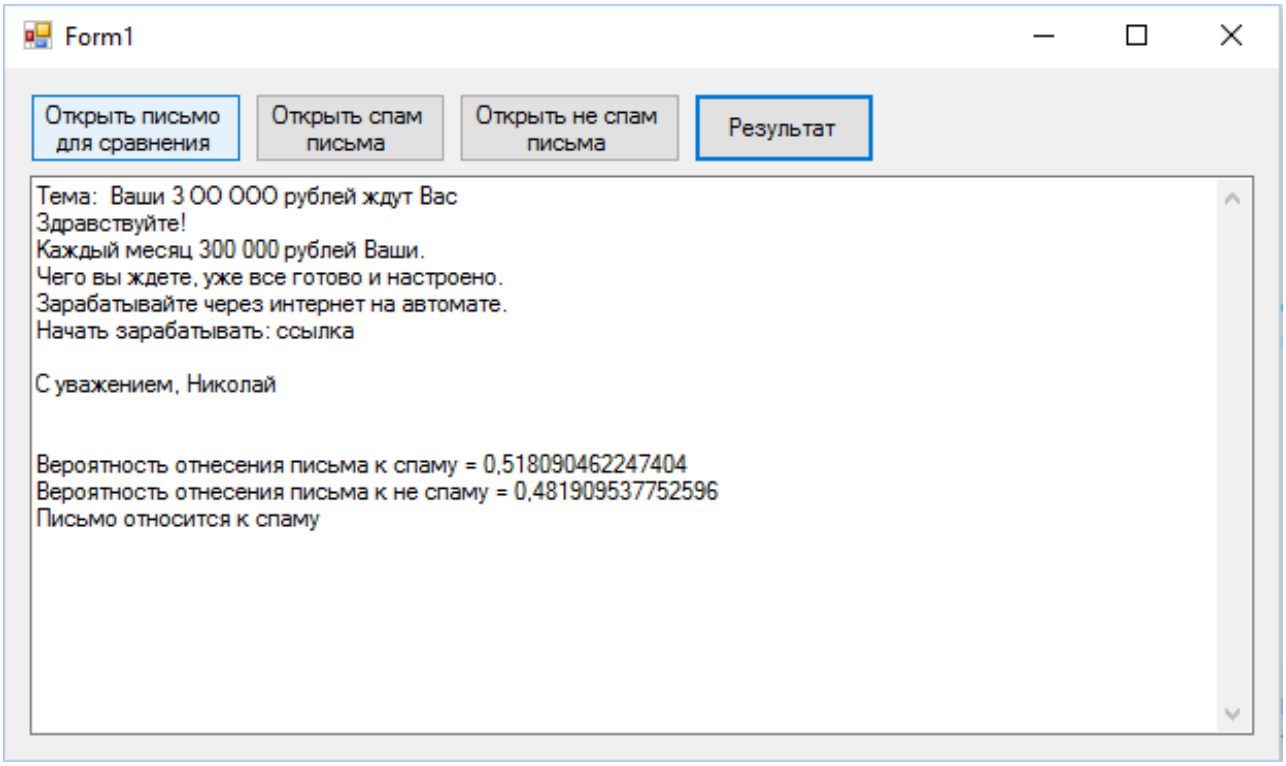

Рисунок 5 – Результат работы мультиномиального наивного байесовского классификатора для письма, которое относится к спаму

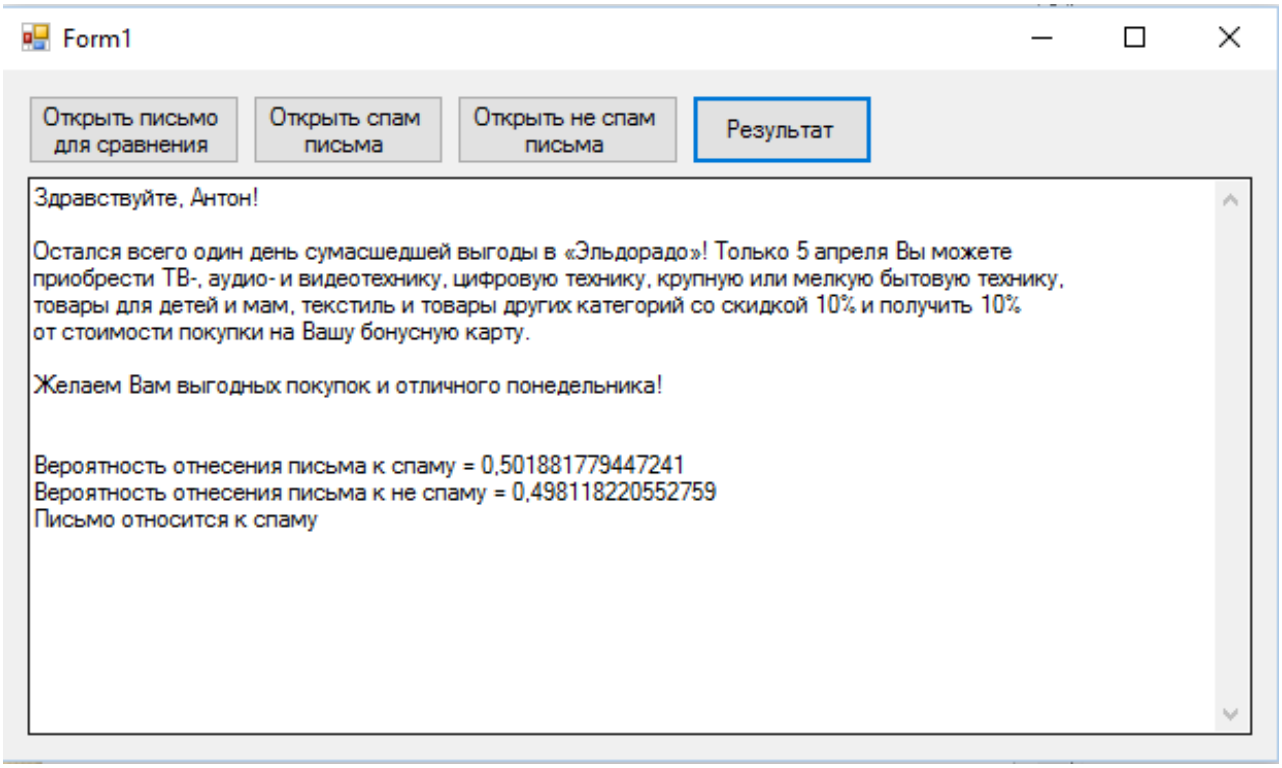

Рисунок 6 – Результат работы мультиномиального наивного байесовского классификатора для письма, которое является не спамом, хотя и содержит некоторые слова, часто встречающиеся в спаме

3.4 Результаты работы программы, решающей практическую задачу методом, основанным на модели «мешка слов» на тестовых письмах

Решение практической задачи методом, основанным на модели «мешка слов» происходило двумя способами. В первом случае на вход подавалось некоторое множество писем и программа должна была разбить их на два кластера, в качестве метода кластеризации использовался метод k-средних. Во втором случае на вход программе, как и предыдущим двум, подавались три различных тестовых письма для проверки: первое письмо являлось не спамом, второе было спамом, третье письмо не являлось спамом, хотя и содержало некоторые слова, часто встречающиеся в спаме.

При запуске программа спрашивает у пользователя какое количество писем ему необходимо распределить. Если это одно письмо, то запускается метод классификации. Если же это несколько писем, то применяется метод кластеризации. Начальный экран программы приведен на Рисунке 7.

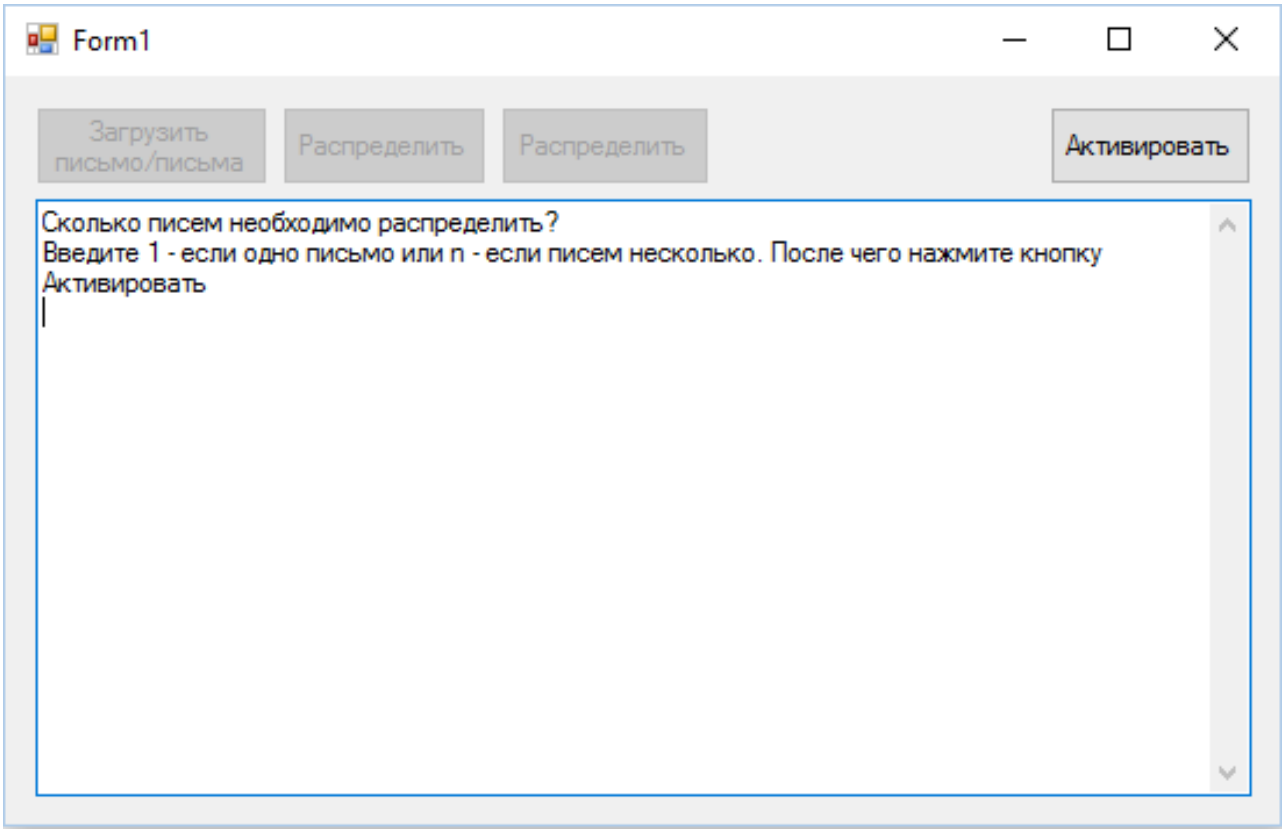

Рисунок 7 – Начальный экран работы программы, основанной на модели «мешка слов»

На Рисунке 9 представлен результат работы данного метода для кластеризации некоторого множества писем. На Рисунке 8 приведен пример одного из множества писем. После проверки результатов оказалось, что только одно письмо было отнесено не к тому классу.

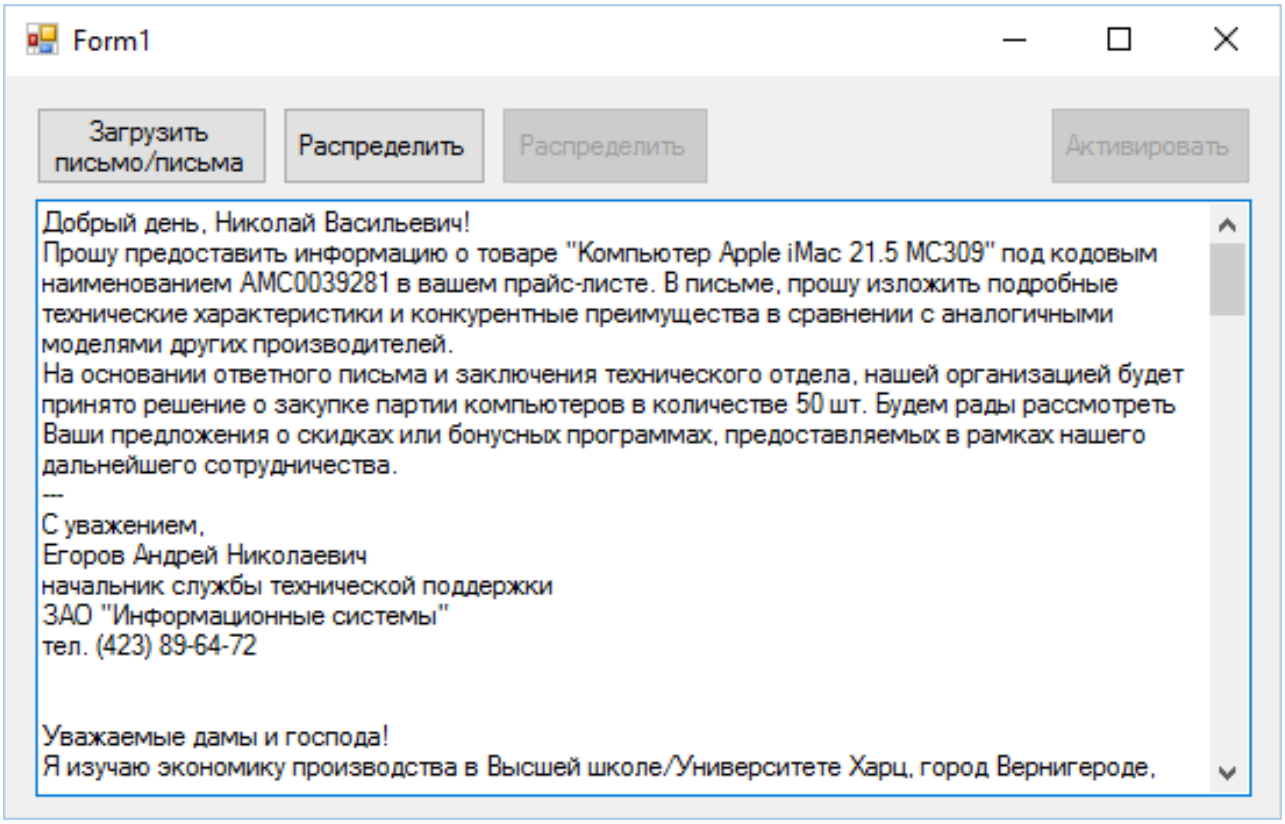

Рисунок 8 – Пример письма из множества писем, поданных на кластеризацию в программу, основанную на модели «мешка слов»

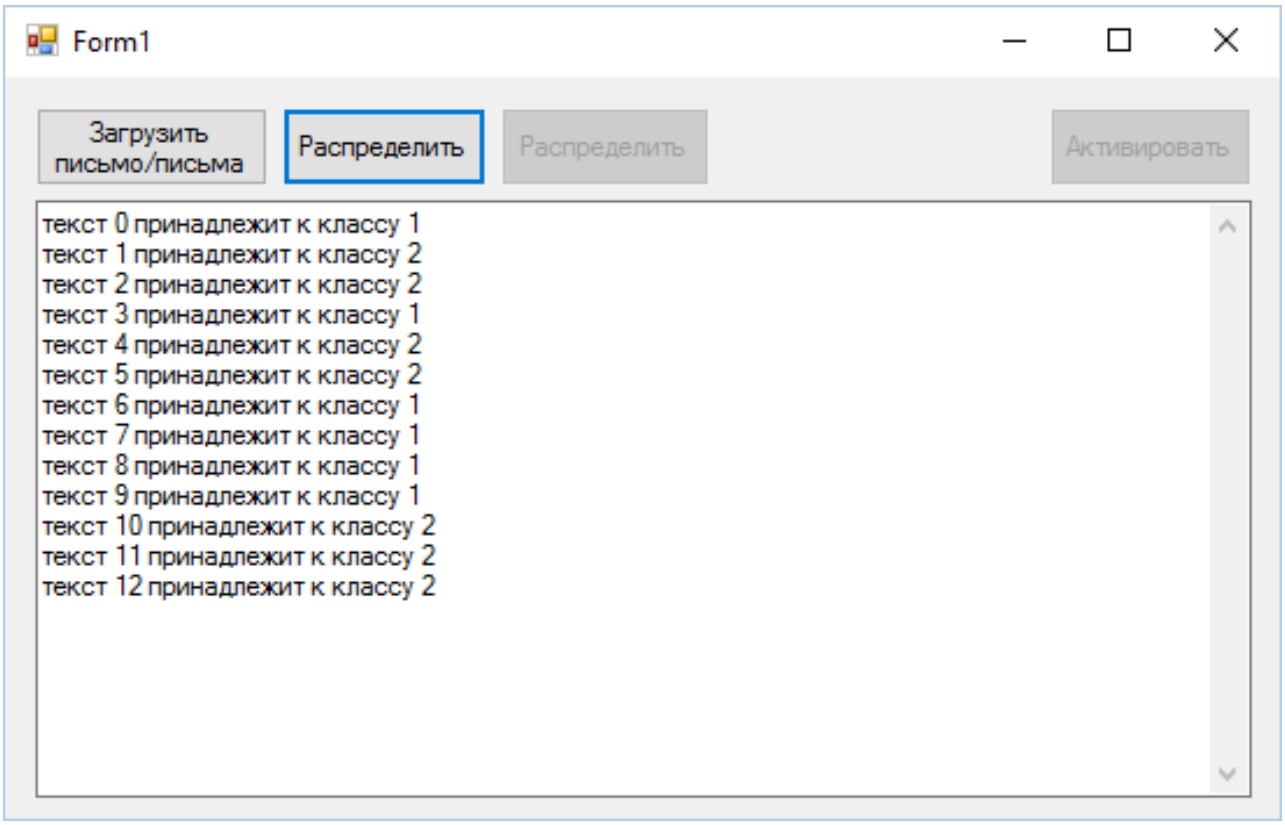

Рисунок 9 – Результат работы программы, основанной на «мешке слов» при кластеризации множества писем

На Рисунках 10-12 представлены результаты работы данного метода для классификации писем. Из этих результатов видно, что письма, поданные на вход, были отнесены к соответствующим классам верно.

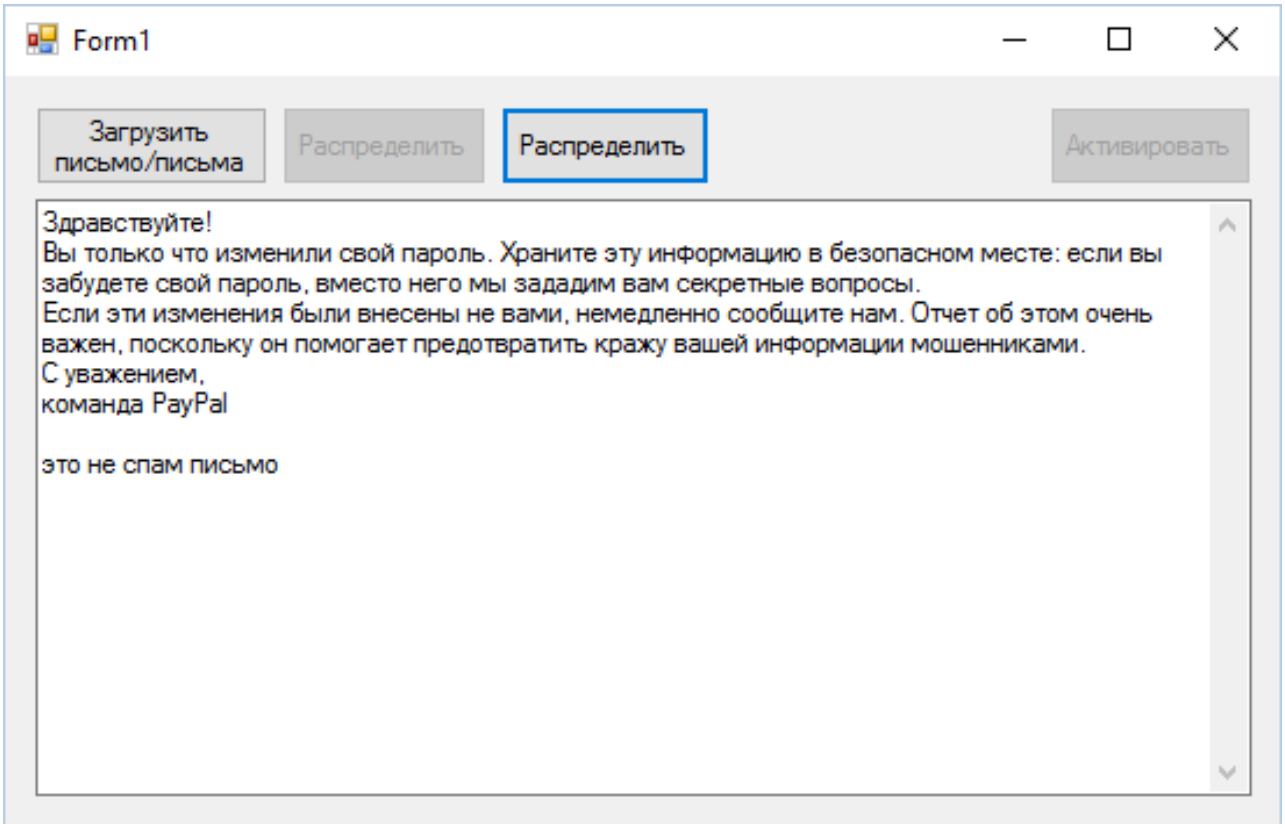

Рисунок 10 – Результат работы метода, основанного на модели «мешка слов» для письма, которое относится к не спаму

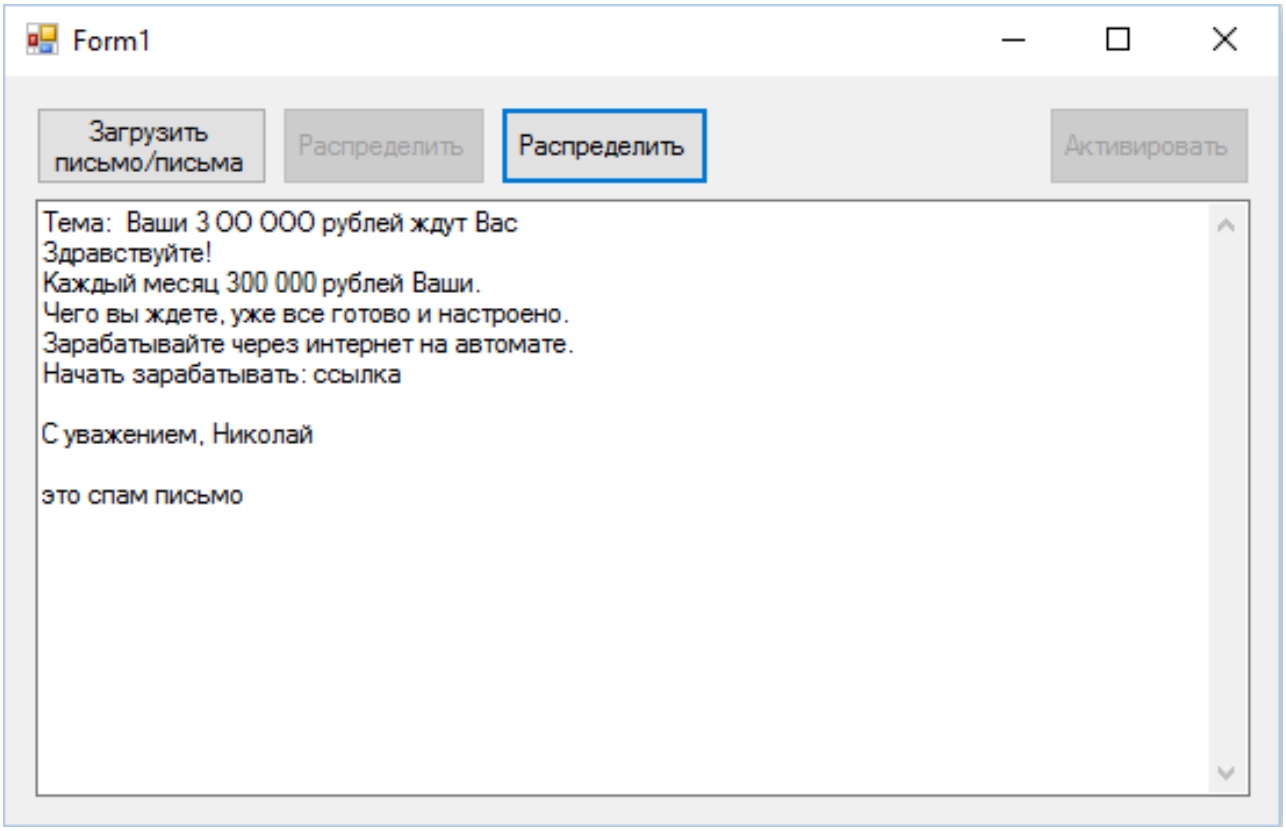

Рисунок 11 – Результат работы метода, основанного на модели «мешка слов» для письма, которое относится к спаму

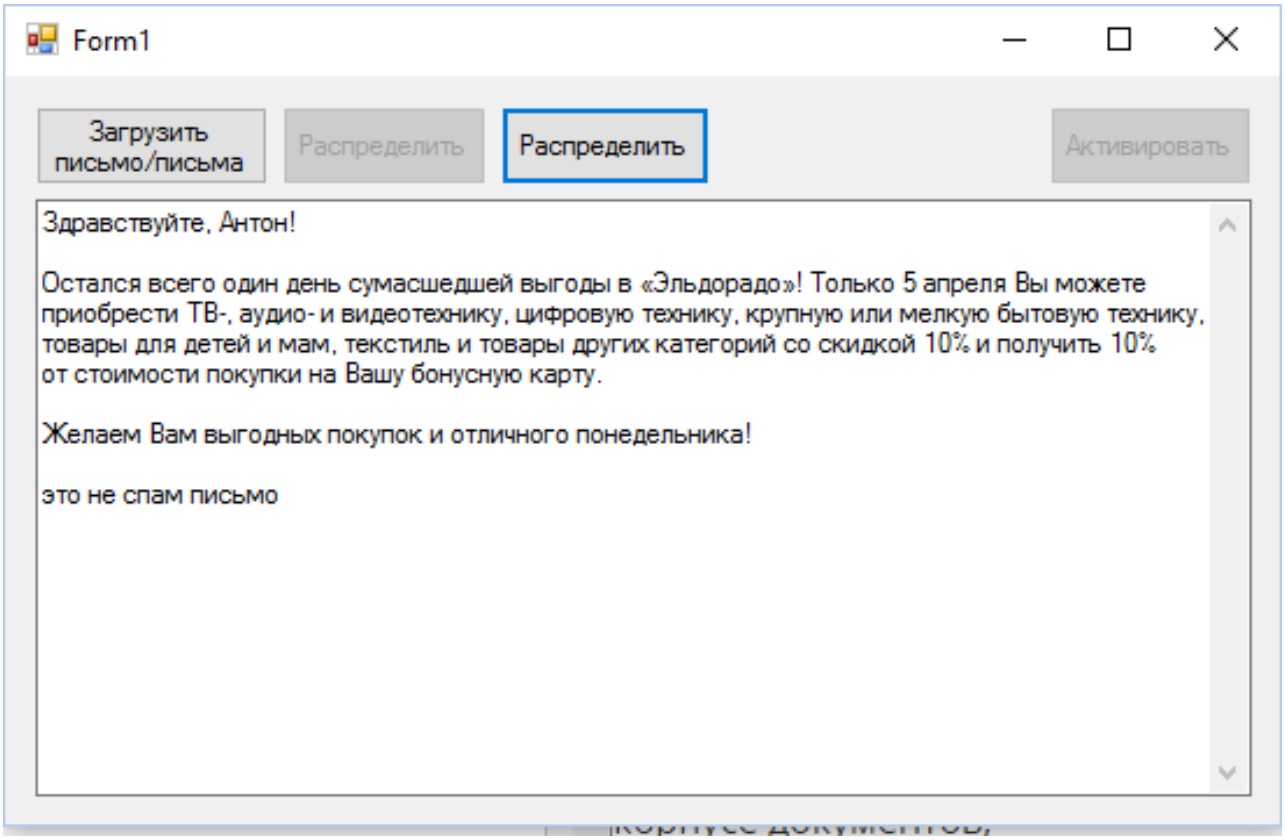

Рисунок 12 – Результат работы метода, основанного на модели «мешка слов» для письма, которое является не спамом, хотя и содержит некоторые слова, часто встречающиеся в спаме

4. Сравнительный анализ методов

4.1 По вычислительной сложности

Рассмотрим вычислительную сложность каждого из алгоритмов:

- В многомерном наивном байесовском классификаторе мы перемножаем вероятности того, что встретилось каждое слово из документа и вероятности того, что не встретилось каждое (словарное) слово, которое не встретилось. Получается модель многомерных испытаний Бернулли, таким образом, вычислительная сложность алгоритма будет равна: $o(n^2)$.
- В мультиномиальном наивном байесовском классификаторе мы перемножаем вероятности того, что мы достали из словаря те самые слова, которые встретились в документе. Считает вероятность принадлежности слова к каждому из классов (спам и не-спам), если оно имеется в базе данных спам слов, в противном случае пропускает это слово, таким образом, вычислительная сложность алгоритма будет равна: $o(n^2)$.
- В методе кластеризации, основанном на мешке слов используется метод k-средних, следовательно вычислительная сложность алгоритма будет равна: $O(nkl)$, где k – число кластеров (в нашей задаче k=2), l –число итераций.
- В методе классификации, основанном на мешке слов вычисляется количество одинаковых слов, путем сравнения каждого слова из обучающей выборки с каждым словом из письма, таким образом, вычислительная сложность алгоритма будет равна: $O(n^3)$.

4.2 По результатам работы

В качестве критерия качества классификации использовалось отношение количества правильно отнесенных писем к общему количеству писем, поданных на вход:

$$Q(f^*, C, \rho) = \frac{P(Cor)}{P(Total)},$$

где $P(Cor)$ – количество верно классифицированных писем, $P(Total)$ – общее количество писем, поданных на вход.

Правильность отнесения письма к определенному классу определяется экспертом.

Каждым методом были обработано около ста пятидесяти писем, которые либо содержали спам, либо нет. Результаты были проанализированы и получены следующие итоги:

- Многомерный наивный байесовский классификатор – точность 80%,
- Мультиномиальный наивный байесовский классификатор – точность 79,3%,
- Классификация на основе модели «мешка слов» – точность 66%.

5. Описание программного приложения

В рамках бакалаврской работы был разработан комплекс программ, реализующий методы решения задачи классификации спама. В программный комплекс включены три основных метода, рассмотренных в предыдущих главах: многомерный наивный байесовский классификатор, мультиномиальный наивный байесовский классификатор и метод классификации, основанный на модели «мешка слов».

Приведем подробное описание состава и характеристик разработанного комплекса.

Указанное программное приложение разработано в бесплатной среде разработки Visual Studio Community 2015. В качестве языка программирования

выбран язык объектно-ориентированного программирования C#, являющийся одним из самых популярных, востребованных и многофункциональных языков программирования. Среда Visual Studio Community 2015 представляет собой бесплатную, полнофункциональную и расширяемую интегрированную среду разработки для создания современных приложений для Windows, Android и iOS, а также веб-приложений и облачных служб.

В начале для каждого метода производилась предварительная подготовка текстов. Удалялись или заменялись все известные и не известные специальные символы, тем самым, для подсчета вероятностей использовались только слова.

5.1 Приложение для многомерного наивного байесовского классификатора

В качестве входных данных для работы алгоритма поступает три файла формата .txt. В первом файле находится письмо, которое необходимо классифицировать, во втором – множество писем, относящихся к спаму, в третьем – множество писем, относящихся к не спаму. Второй и третий файлы являются обучающей выборкой. На Рисунке 13 представлена форма данной программы.

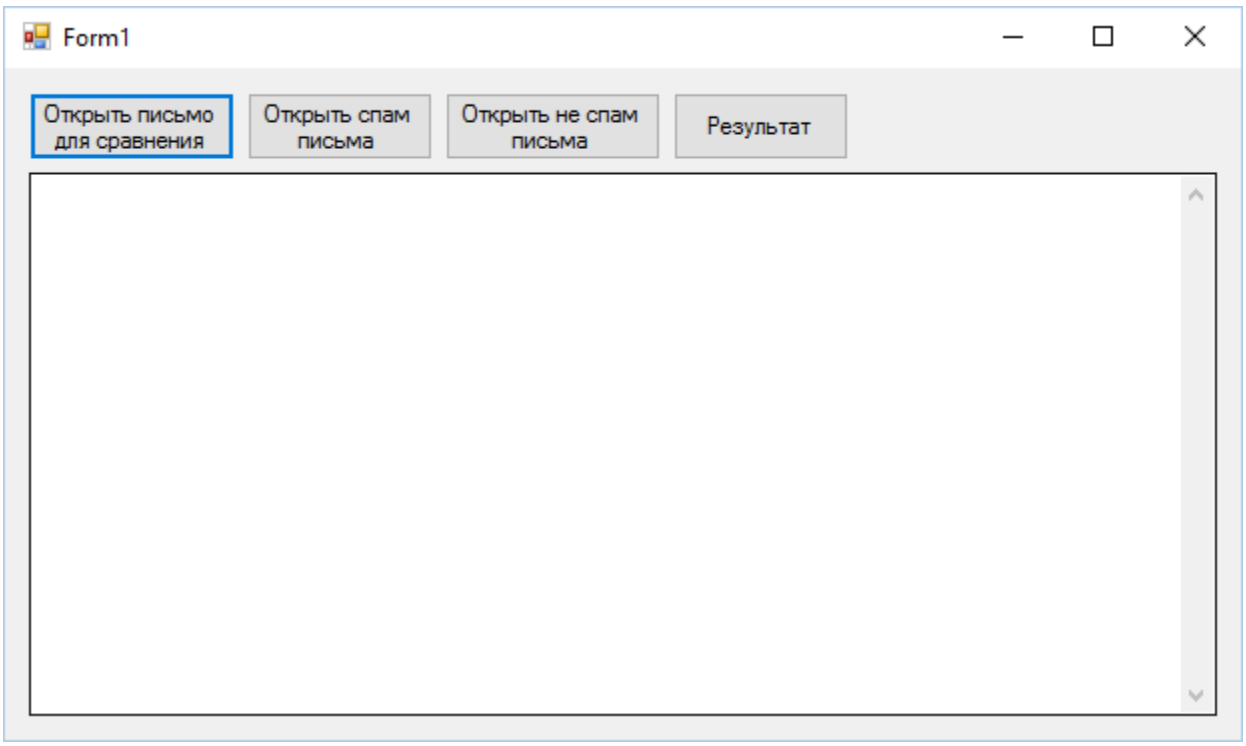

Рисунок 13 – Форма приложения для многомерного наивного байесовского классификатора

Кнопка *Открыть письмо для сравнения* позволяет загрузить файл, в котором находится письмо, требующее классификации. На рисунке 14 демонстрируется окно выбора файла для классификации.

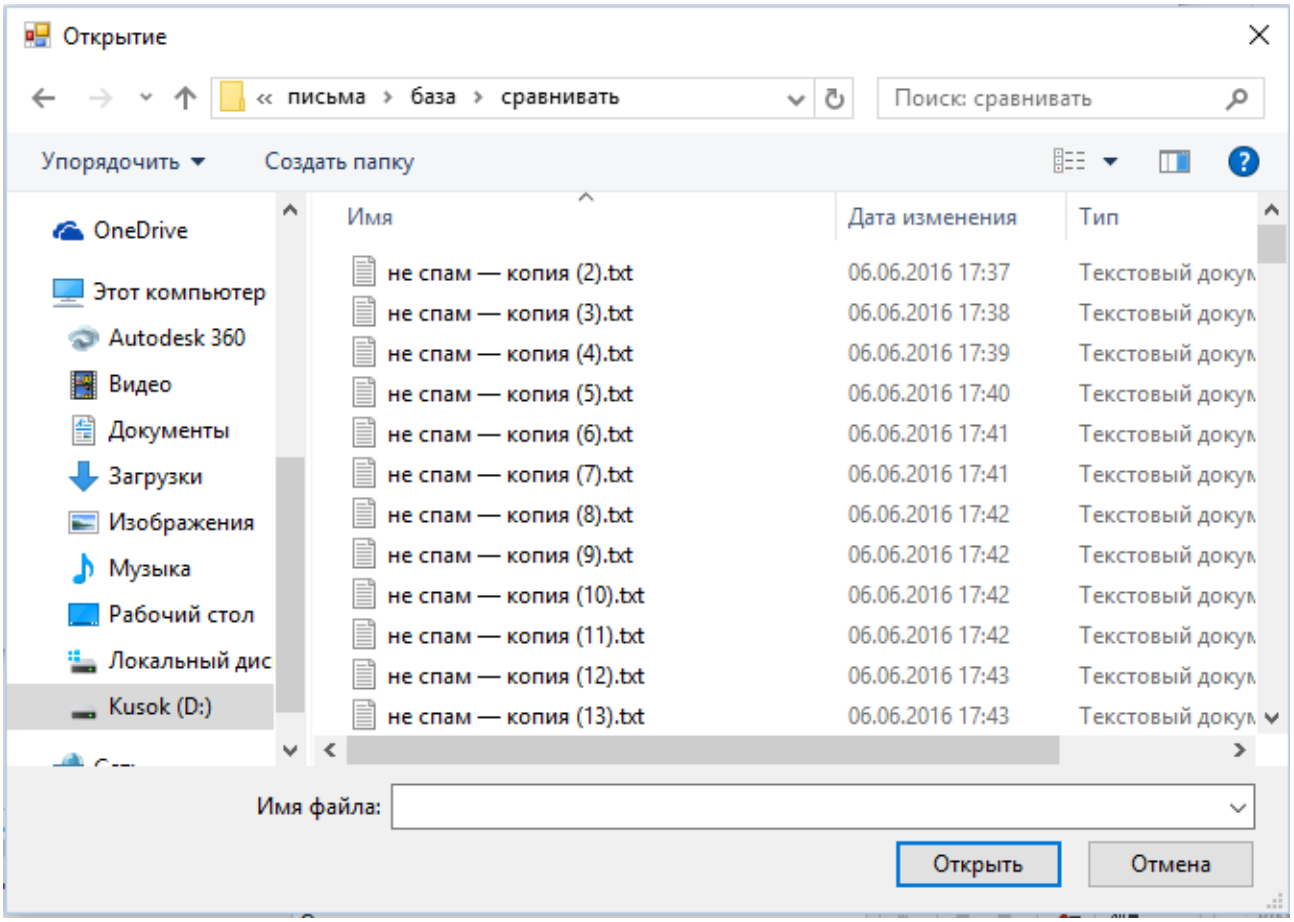

Рисунок 14 – Окно выбора файла

Кнопка *Открыть спам письма* позволяет загрузить файл, в котором находится множество писем, относящихся к спаму. Результат открытия обучающей выборки для спама показан на рисунке 15.

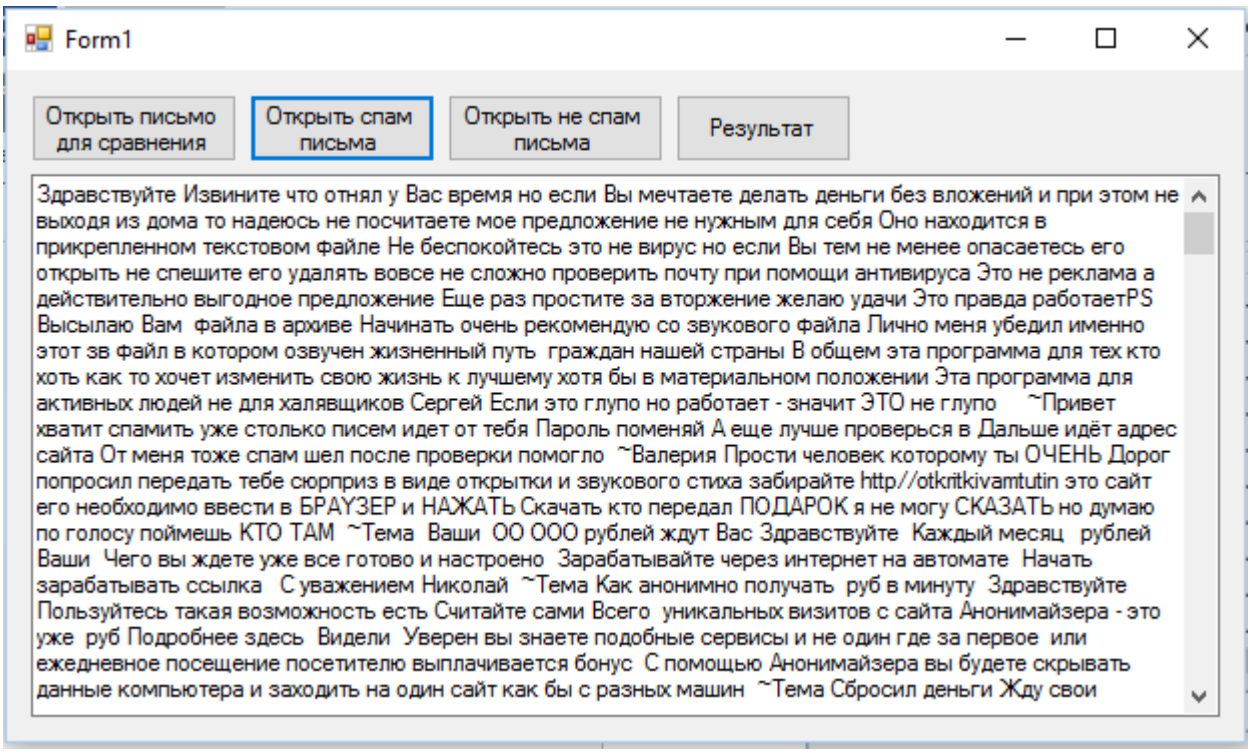

Рисунок 15 – Результат открытия выборки спама

Кнопка *Открыть не спам письма* позволяет загрузить файл, в котором находится множество писем, относящихся к не спаму. Результат открытия обучающей выборки для не спама показан на рисунке 16.

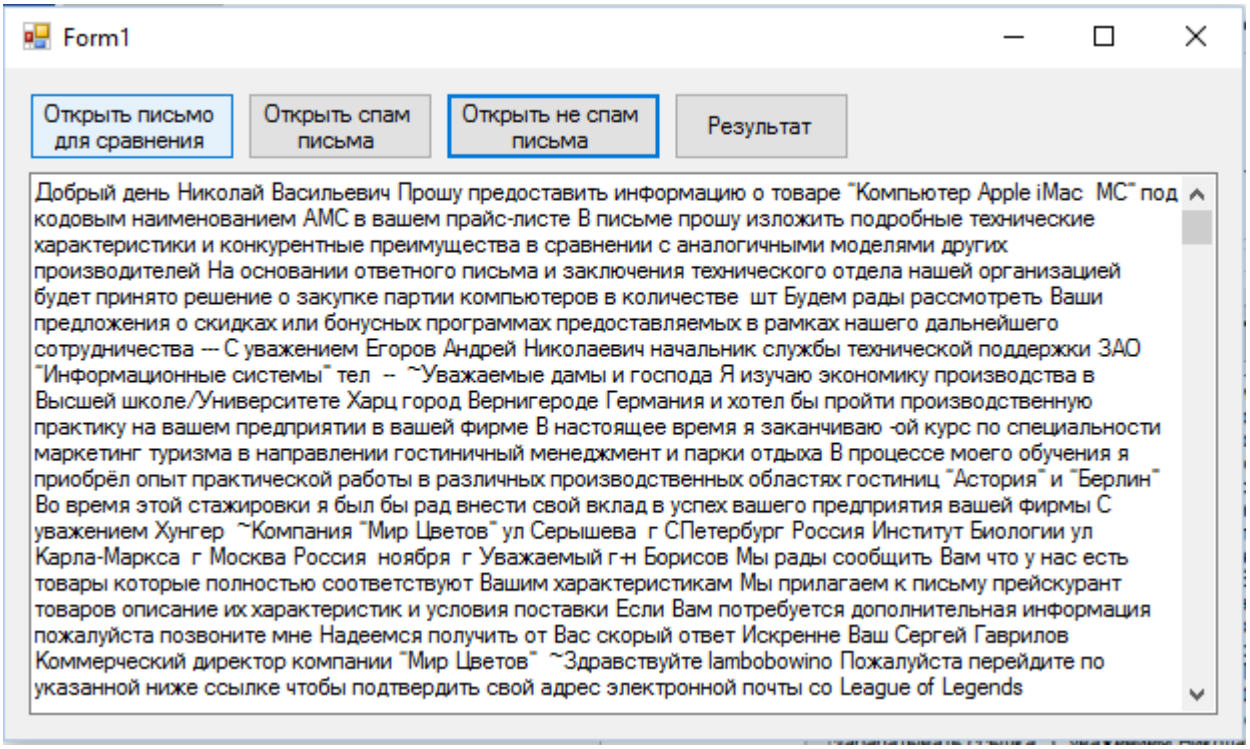

Рисунок 16 – Результат открытия выборки не спама

По кнопке *Результат* производятся все необходимые вычисления, и в TextBox выводится результат классификации. Результат классификации представлен на рисунке 17.

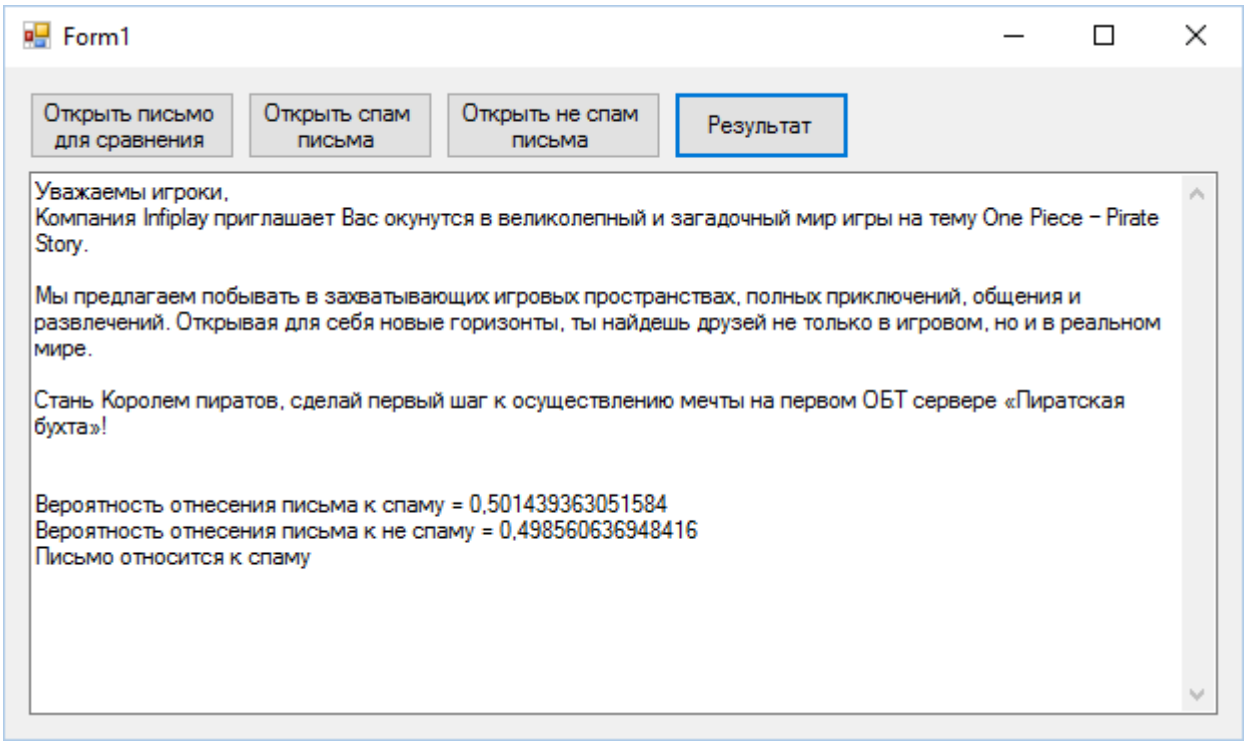

Рисунок 17 – Результат классификации

5.2 Приложение для многомерного наивного байесовского классификатора

В качестве входных данных для работы алгоритма поступает три файла формата .txt. В первом файле находится письмо, которое необходимо классифицировать, во втором – множество писем, относящихся к спаму, в третьем – множество писем, относящихся к не спаму. Второй и третий файлы являются обучающей выборкой. На Рисунке 18 представлена форма данной программы.

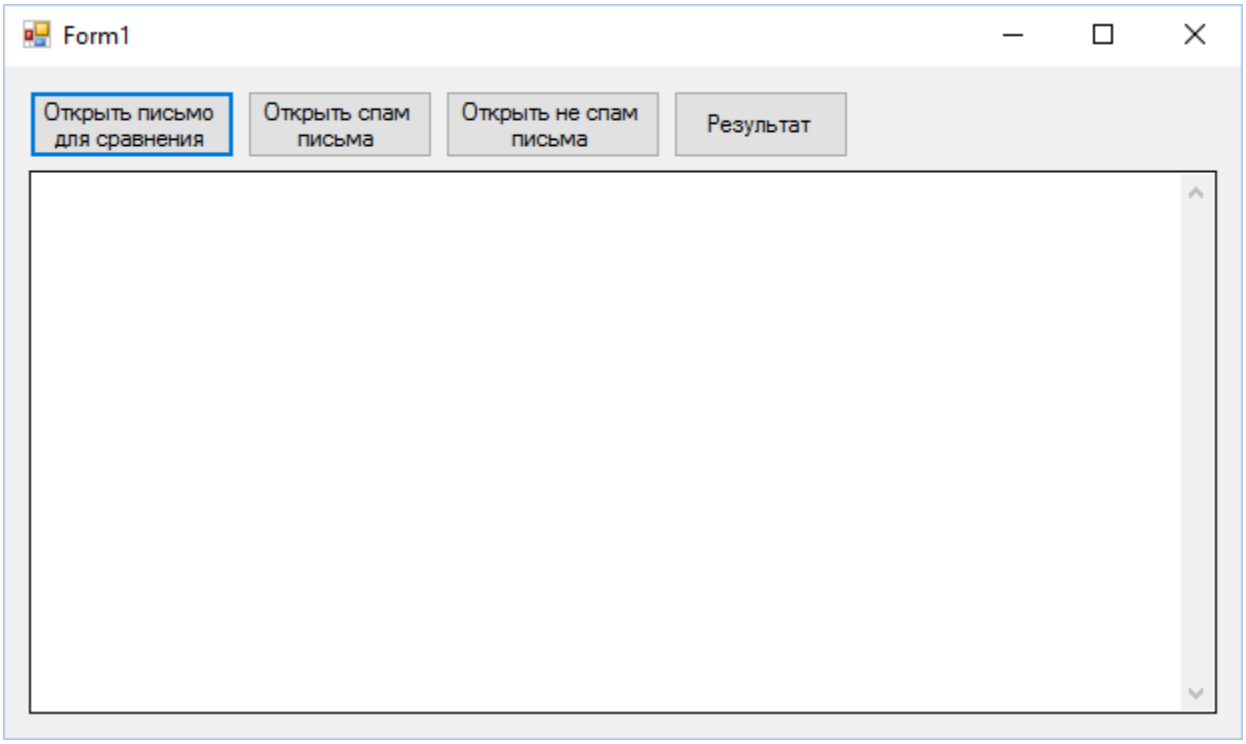

Рисунок 18 – Форма приложения для мультиномиального наивного байесовского классификатора

По первым трем кнопкам происходит все то же самое, что и в приложении для многомерного наивного байесовского классификатора.

Различие данных программ в производимых вычислениях по кнопке *Результат*.

5.3 Приложение для многомерного наивного байесовского классификатора

В качестве входных данных для работы алгоритма поступает файл формата .txt. В данном файле находится либо одно письмо, которое необходимо классифицировать, либо множество писем, которое необходимо разбить на кластеры. На Рисунке 19 представлена форма данной программы.

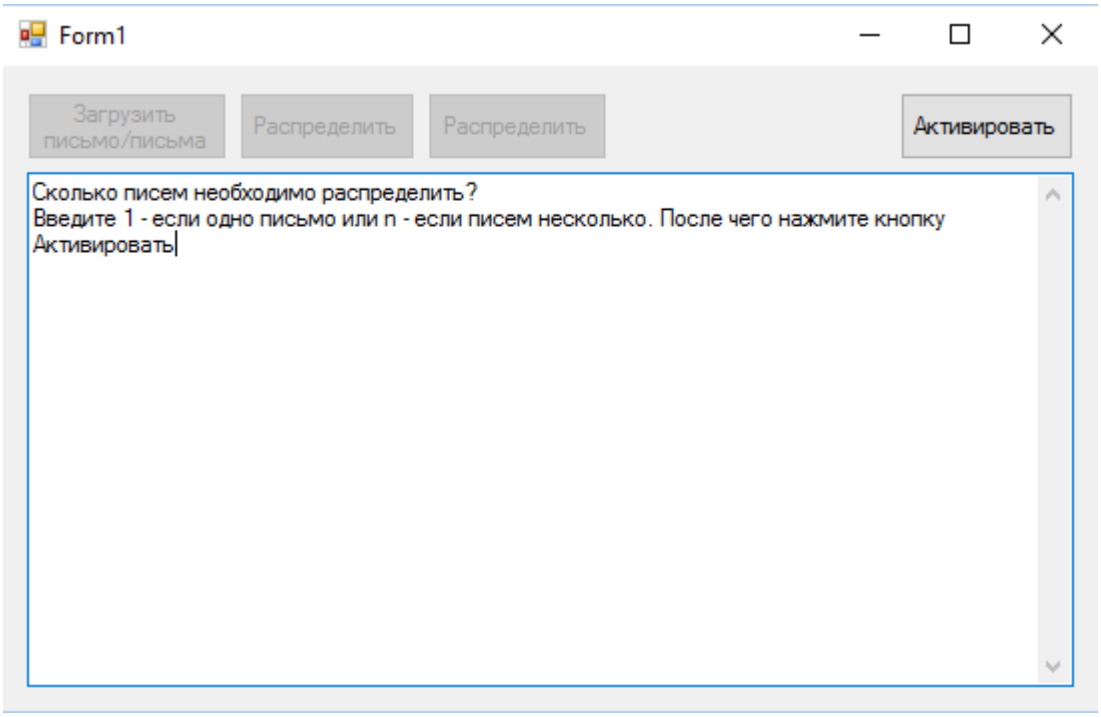

Рисунок 19 – Форма приложения для метода, основанного на «мешке слов»

Первое что происходит при открытии приложения – это вывод инструкции для пользователя.

Кнопка *Активировать* активирует необходимые кнопки основываясь на введенных пользователем данных.

Кнопка *Загрузить письмо/письма* активируется в любом случае, она позволяет открыть файл с письмами.

Первая кнопка *Распределить* активируется при необходимости кластеризации множества писем. При нажатии на нее производятся вычисления и выводится результат в TextBox. Результат работы данного метода для множества писем представлен на рисунке 20.

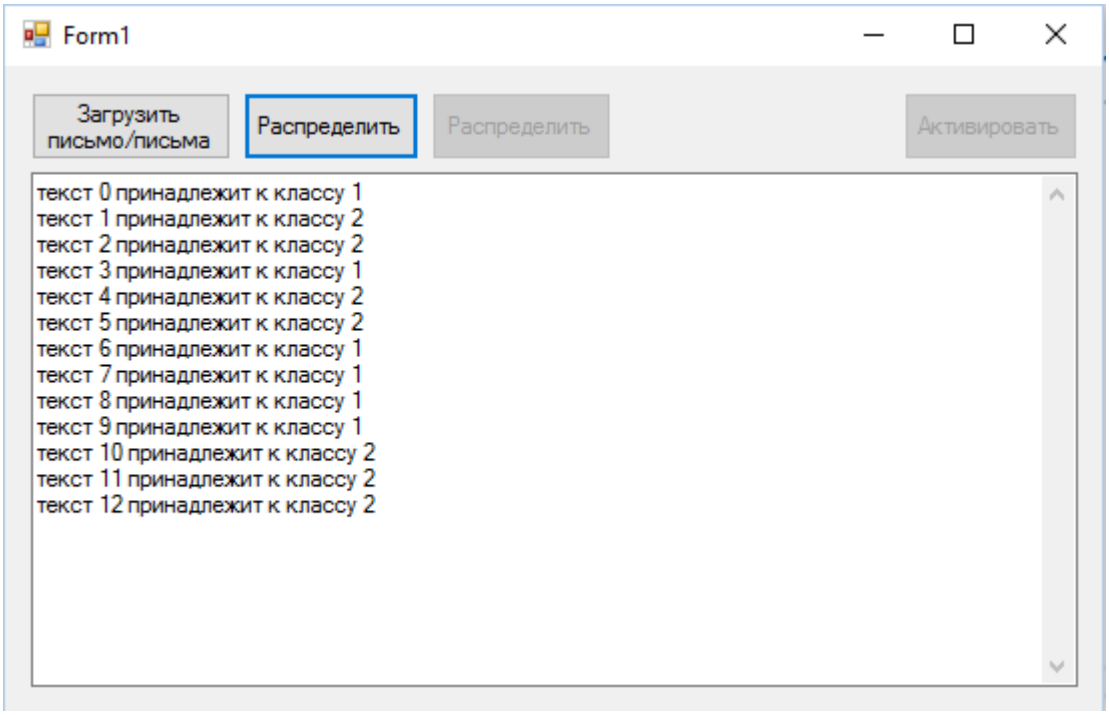

Рисунок 20 – Результат работы для множества писем

Вторая кнопка *Распределить* активируется при необходимости классификации одного письма. При нажатии на нее производится загрузка обучающей выборки, производятся вычисления и выводится результат в TextBox. Результат работы данного метода для одного письма представлен на рисунке 21.

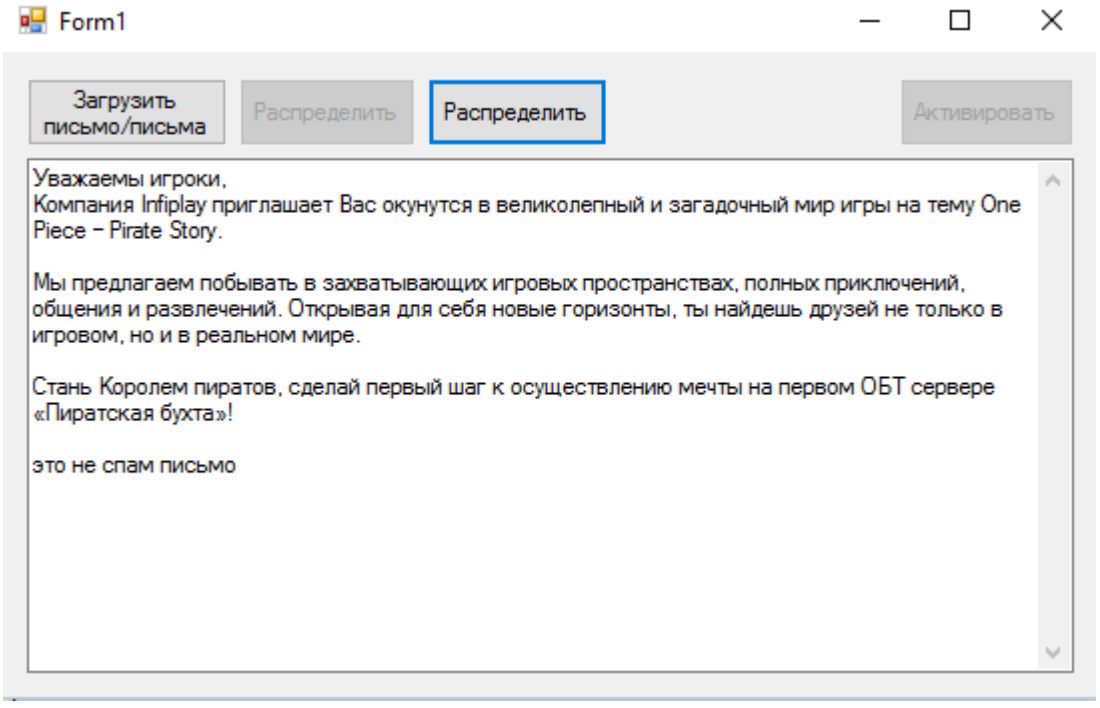

Рисунок 21 – Результат работы для одного письма

ЗАКЛЮЧЕНИЕ

В работе получены следующие результаты:

- Изучены методы фильтрации спама: байесовский классификатор на основе многомерной и мультиномиальной модели.
- Разработан метод фильтрации спама – метод классификации на основе модели мешка слов.
- Создано программное приложение, реализующее вышеперечисленные алгоритмы фильтрации спама.
- Проведены вычислительные эксперименты по сравнению вычислительной сложности реализованных методов и результатов их работы.
- Решена практическая задача фильтрации спама.
- Проведен анализ результатов решения практической задачи, полученных рассматриваемыми методами.

Результаты работы опубликованы на международной научно-технической конференции студентов, аспирантов и молодых ученых «Проспект Свободный – 2015» (Красноярск, 2015), международной научно-технической конференции студентов, аспирантов и молодых ученых «Проспект Свободный – 2016» (Красноярск, 2016).

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Воронцов, К. В. Лекции по алгоритмам кластеризации и многомерного шкалирования / К. В. Воронцов. Москва: МГУ, 2007. – 18 с.
2. Дюран, Б. Кластерный анализ: пер. с англ. Е. З. Демиденко под ред. А. Я. Боярского / Б. Дюран, П. Одел. Москва: «Статистика», 1977. – 128 с.
3. Загоруйко, Н. Г. Прикладные методы анализа данных и знаний / Н. Г. Загоруйко. Новосибирск: ИМ СО РАН, 1999. – 270 с.
4. Лепский, А. Е. Математические методы распознавания образов / А. Е. Лепский, А. Г. Броневич. Таганрог: Изд-во ТТИ ЮФУ, 2009. – 155 с.
5. Местецкий, Л. М. Математические методы распознавания образов / Л. М. Местецкий. Москва: МГУ, 2002. – 139 с.
6. Миркин, Б. Г. Методы кластер – анализа для поддержки принятия решений: обзор: препринт WP7/2011/03/ Б. Г. Миркин. Москва: Изд. Дом Национального исследовательского университета «Высшая школа экономики», 2011. – 88 с.
7. Мэннинг, К. Введение в информационный поиск / К. Мэннинг, П. Рагхаван, Х. Шютце. Москва: Вильямс, 2011. – 163 с.
8. Потапов, А. С. Распознавание образов и машинное восприятие / А. С. Потапов. Москва: "Политехника", 2007. – 552 с.
9. McCallum, A. A comparison of event models for naive bayes text classifcation / A. McCallum, K. Nigam. AAAI-98 workshop on learning for text categorization, 1998. – 41 с.
10. Ветров, Д. П. Спецкурс «Байесовские методы машинного обучения». Лекция 2 «Вероятностная постановка задач классификации и регрессии. Байесовские решающие правила. Обобщенные линейные модели» / Д. П. Ветров, Д. А. Кропотов. [Электронный ресурс] - Режим доступа: http://www.machinelearning.ru/wiki/images/7/78/BayesML-2009-2a.pdf
11. Кулистов, А. В. Задача кластеризации данных и её решение методами к-средних и Ланса-Уильямса / А. В. Кулистов // Сборник материалов

международной конференции студентов, аспирантов и молодых ученых «Проспект Свободный-2015», г. Красноярск, СФУ, 2015. – с. 24-28.

12. Кулистов, А. В. Методы фильтрации спама / А. В. Кулистов // Сборник материалов международной научно-технической конференция студентов, аспирантов и молодых учёных «Проспект Свободный-2016» г. Красноярск, 2016. – 31 с.

ПРИЛОЖЕНИЕ А

Код программы, реализующий многомерный наивный байесовский классификатор

Страница изъята

Страница изъята

Страница изъята

Страница изъята

Страница изъята

Страница изъята

Страница изъята

Страница изъята

ПРИЛОЖЕНИЕ Б

Код программы, реализующий мультиномиальный наивный байесовский классификатор

Страница изъята

Страница изъята

Страница изъята

Страница изъята

Страница изъята

Страница изъята

Страница изъята

ПРИЛОЖЕНИЕ В

Код программы, реализующий кластеризацию и классификацию на основе модели «мешка слов»

Страница изъята

Страница изъята

Страница изъята

Страница изъята

Страница изъята

Страница изъята

Страница изъята

Страница изъята

Страница изъята

Страница изъята