МИНИСТЕРСТВО ВЫСШЕГО И СРЕДНЕГО СПЕЦИАЛЬНОГО ОБРАЗОВАНИЯ РЕСПУБЛИКИ УЗБЕКИСТАН

САМАРКАНДСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ имени АЛИШЕРА НАВОИ

На правах рукописи
УДК 519.681.5

ХИММАТОВ ИБОДИЛЛА

РАЗРАБОТКА АЛГОРИТМОВ И ПРОГРАММНЫХ СРЕДСТВ КОНТРОЛЯ ДОСТОВЕРНОСТИ ОБРАБОТКИ ЭЛЕКТРОННЫХ ТЕКСТОВ НА ОСНОВЕ МЕТОДОВ УПРАВЛЕНИЯ МОРФОЛОГИЧЕСКИМ СЛОВАРЕМ

МОРФОЛОГИК ЛУҒАТНИ БОШҚАРИШ АСОСИДА ЭЛЕКТРОН МАТНЛАРГА ИШЛОВ БЕРИШ ИШОНЧЛИГИНИ НАЗОРАТ ҚИЛИШ АЛГОРИТМЛАРИ ВА ДАСТУРИЙ ВОСИТАЛАРИНИ ИШЛАБ ЧИҚИШ

Специальность 5А110701 – Информационные технологии в образовании

Диссертация
на соискание академической степени магистра

Научный руководитель
доц. Ахатов А.Р.

САМАРКАНД – 2016

МИНИСТЕРСТВО ВЫСШЕГО И СРЕДНЕГО СПЕЦИАЛЬНОГО ОБРАЗОВАНИЯ
РЕСПУБЛИКИ УЗБЕКИСТАН
САМАРКАНДСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

Факультет: Механика-математика

Студент магистратуры: Химматов Ибодилла

Кафедра: информационные технологии

Научный руководитель: доц. А.Р.Ахатов

Учебный год: 2015-2016

Специальность: 5А110701 –
Информационные технологии в образовании

АННОТАЦИЯ МАГИСТЕРСКОЙ ДИССЕРТАЦИИ

Актуальность темы. При вводе, передаче и обработке информации на ЭВМ могут возникать различного рода ошибки в текстах. Причинами ошибок могут быть: устройства сканирования, распознающие программные системы, человек-оператор, обрабатывающий эту информацию; различные помехи, влияющие на канал связи; сбои и отказы электронных оборудований. Значимость проблемы контроля достоверности информации подчеркивается еще и тем, что вся деловая информация в нашей республике ведется на узбекском языке, однако, не все граждане республики обладают умением правописания узбекских текстов, в результате чего в текстах появляются дополнительные орфографические ошибки. Следовательно, актуальными являются задачи разработки методов, моделей, алгоритмов и программной системы обработки текстов на узбекском языке для контроля и коррекции ошибок, а также практическая ее реализация в системе ЭСД предприятий и организаций

Цель работы. Разработка алгоритмов и программных средств контроля текстовой информации на базе моделей управления морфологическим словарем, демонстрирующих возможность эффективного решения задачи морфологического анализа, как ключевого этапа обеспечения достоверности электронных текстов.

Задачи исследований. Изучение лингвистических стратегий и правил, отвечающих словообразовательным законам языка. Поиск случаев применения описываемых грамматических конструкций в корпусе текстов. Разработка способов морфологического анализа без словаря и методики анализа словоформ в морфологической модели. Разработка алгоритма морфологического анализа на основе индексирования и проектирование словарной морфологии. Обоснование сложности машинного анализа узбекского предложения, разработка методики постморфологического анализа. Разработка общей схемы действий сегментационного анализа и внутрисегментного анализа. Построение архитектуры и программно-реализуемых модулей системы контроля и исправления орфографических ошибок на основе моделей управления морфологическим словарем.

Объект исследования – тексты и документы электронной системы документооборота (ЭСД) предприятий и организаций. **Предмет исследования** – методы, алгоритмы и система контроля достоверности текстовой информации.

Методика и методы исследования: Основы построения систем управления, теория алгоритмизации, теории передачи и обработки информации, теория программирования, лексикологии узбекского, тюркского языков.

Научная новизна работы состоит в том, что предложенные алгоритмы позволяют создать модуль морфологического анализа и реализовать разработанные методы предсказания основы слова и части речи, а также самообучающего анализатора в системе контроля орфографических ошибок для электронных текстов на узбекском языке.

Научная и практическая значимость результатов исследования. Результаты диссертационной работы могут быть применены при создании систем контроля и коррекции орфографии узбекского языка, генерации и синтеза текста из речи, машинного перевода текстов

на узбекском языке. Также систему возможно использовать при обучении основам грамматики узбекского языка.

Методы и программная система контроля орфографических ошибок, разработанные на основе многоуровневой модели представления словоформ узбекского языка не требуют глубокого знания языка и решают задачи при небольшом объеме словаря, работают с оптимальной скоростью, потребляют приемлемое количество памяти.

Структура и объем работы: Диссертация состоит из введения, трех глав, заключения, списка использованной литературы и приложений. Работа изложена на 70 страницах основного текста, проиллюстрирована 13 рисунками. Использованы 59 источников литературы.

Основные результаты работы.

В диссертационной работе рассмотрены вопросы реализации алгоритмов контроля достоверности электронных текстов на узбекском языке. Предложен подход к созданию системы выделения лемм из электронных текстов на основе статистических данных. Наличие словаря лемм упрощает морфологический анализ слов, повышает точность. Каждой лемме в словаре возможно добавление перевода, что позволит построить машинный переводчик.

Разработаны методики определения объема избыточности, обеспечивающего требуемое качество контроля текстов и расчета рационального объема памяти программной системы обработки информации для контроля и коррекции ошибок в текстах на естественных языках, в частности, на узбекском.

Исследованы правила описания формального анализа узбекских словоформ, на основе чего разработаны модель морфологического анализа, обобщенный алгоритм построения и структура программной системы для контроля и коррекции ошибок, не требующей от пользователя глубокого знания языка и позволяющей проводить контроль ошибок с ограниченным объемом словаря словоформ.

Обоснован выбор программной среды для реализации системы контроля достоверности электронных текстов. Проанализированы основные возможности языка программирования VB.NET. Использованы широкие возможности работы с текстом, структурами, которые предоставляет VB.NET, что позволило в значительной мере увеличить общую производительность системы.

Разработаны и реализованы программные модули проектирования базы данных морфологического словаря. Реализована структура анализатора, которая имеет завершенный логический смысл, выполняет необходимую обработку и возвращает унифицированный набор результатов данных.

Определены функции и эксплутационные требования к программному обеспечению систем машинного анализа. На различных примерах исходных данных получены результаты обработки морфологического лексикона.

Частными результатами исследований и практических разработок являются следующие:

- разработаны методы морфологического анализа применительно к узбекскому языку;
- созданы методы предсказания основы слова и части речи для существительных и глаголов;
- разработаны алгоритмы реализации и выполнено программирование системы на языке VB.NET;
- проведён анализ производительности системы, результаты которого удовлетворяют требованиям к качеству передачи и обработки информации.

Студент магистратуры И.Химматов

Научный руководитель доц. А.Р.Ахатов

ЎЗБЕКИСТОН РЕСПУБЛИКАСИ
ОЛИЙ ВА ЎРТА МАХСУС ТАЪЛИМ ВАЗИРЛИГИ
АЛИШЕР НАВОИЙ НОМИДАГИ
САМАРҚАНД ДАВЛАТ УНИВЕРСИТЕТИ

Факультет: Механика-математика

Магистратура талабаси: Химматов Ибодилла

Кафедра: ахборотлаштириш технологиялари

Илмий раҳбар: доц. А.Р.Ахатов

Ўқув йили: 2015-2016

Мутахассислиги: 5А110701 – Таълимда ахборот технологиялари

МАГИСТРЛИК ДИССЕРТАЦИЯСИНИНГ АННОТАЦИЯСИ

Мавзунинг долзарблиги. ЭҲМларда маълумотларни киритиш, қайта ишлаш ва узатиш жараёнларида турли табиатдаги хатолар пайдо бўлади. Хатоларнинг манбалари сканерлаш ускуналари, танувчи дастурий тизимлар, маълумотларга ишлов берувчи инсон-оператор, алоқа каналларига таъсир кўрсатаётган шовқинлар, электрон воситалар ишдан чиқиши бўлиши мумкин. Маълумотлар ишончлигини назорат қилиш муаммосининг муҳимлигини республикамизда барча иш хўжжатлари ўзбек тилида юритилиши, аммо ўзбек тили имлосини барча фуқаролар (аҳоли кўп миллатлилиги туфайли) ҳам мукаммал ўзлаштира олмаслиги, натижада эса қайта ишланаётган матнли маълумотлар таркибида қўшимча орфографик хатолар пайдо бўлиши кўрсатади. Шу туфайли, ўзбек тилида бериладиган матнларда хатоларни назорат ва тахрир қилувчи модел, услуб, алгоритм ва дастурий тизимни ишлаб чиқиш долзарб тадқиқотлар мавзуси ҳисобланади.

Ишнинг мақсади ва вазифалари. Матнли маълумотларни назорат қилиш алгоритм ва дастурий воситаларни электрон матнлар ишончлигини асосий босқичи бўлган морфологик таҳлил маслаларининг самарали ечими имкониятини намойиш қилувчи морфологик луғатни бошқариш моделлари асосида ишлаб чиқиш.

Қўйилган мақсадга эришиш учун қуйидаги **масалалар ечилган**. Тилнинг сўз шакллантирувчи қонунларига жавоб берувчи стратегия ва қоидаларни ўрганиш. Тавсифланадиган грамматик конструкцияларни матн корпусида қўллаш холатларини излаш. Луғатсиз таҳлил қилиш усулларни ва морфологик моделда сўз шаклларини таҳлил қилиш услубиятини ишлаб чиқиш. Морфологик таҳлил алгоритмини индекслаш асосида ишлаб чиқиш ва луғатли морфологияни лойиҳалаш. Ўзбек тилидаги гап таҳлилининг мураккаблигини асослаш, постморфологик таҳлил услубиятини ишлаб чиқиш. Сегментацион ва сегмент ичи таҳлилдаги характаларнинг умумий схемасини ишлаб чиқиш. Морфологик луғатни бошқариш моделлари асосида орфографик хатоларни назорат қилиш ва тузатиш тизимининг архитектураси ва дастурий-жорийлашган модулларини ишлаб чиқиш.

Тадқиқот объекти – ташкилот ва муассасаларнинг электрон хўжжат алмашув тизими матнлари ва хўжжатлари. **Тадқиқот предмети** – матнли информация ишончлигини назорат қилувчи услуб, алгоритм ва тизимни ишлаб чиқиш.

Тадқиқот услубияти ва услублари. Бошқарув тизимларни қуриш назарияси, алгоритмлаш назарияси, маълумотларни узатиш ва қайта ишлаш назарияси, дастурлаш назарияси, ўзбек ва турк тилларининг лесикологияси.

Тадқиқот натижаларининг илмий жиҳатдан янгилик даражаси. Таклиф қилинган алгоритмлар морфологик таҳлил модулини яратиш ва ишлаб чиқилган сўз туркумларнинг ва сўзларнинг негизларини башорат қилиш усуллари, ҳамда ўзбек тилида берилган электрон матнлар учун орфографик хатоларни назорат қилиш тизимида ўз-ўзини ўргатувчи анализаторни амалга ошириши имконини беради.

Тадқиқот натижаларининг амалий аҳамияти ва татбиқи. Диссертация иши натижалари ўзбек тили орфографиясини назорат қилиш ва таҳрирлаш, нутқдан матнни генерациялаш ва синтезлаш, ўзбек тилидаги матнларни машинали таржима тизимларни яратишда қўлланиши мумкин. Шу қаторда тизимдан ўзбек тили грамматикасини ўргатишда фойдаланиш мумкин.

Ўзбек тили сўз шаклларини тавсифлашнинг кўп босқичли модели асосида ишлаб чиқилган орфографик хатоларни назорат қилиш услуб ва дастурий тизими фойдаланувчидан тилни чуқур билишни талаб этмайди, масалаларни катта бўлмаган ҳажмдаги лўғат билан ечади, оптимал тезликда ишлайди ва фаолияти даврида мақбул ҳажмдаги хотирани эгаллайди.

Иш тузилиши ва таркиби. Диссертация кириш, уч боб, ҳулоса қисми, фойдаланилган адабиётлар руйхати ва иловалардан иборат. Ишнинг асосий матни - 70 бет, расмлар - 13. Ишда 59 адабиёт манбалари келтирилган.

Бажарилган ишнинг асосий натижалари. Хулоса ва таклифларнинг қисқача умумлаштирилган ифодаси.

Диссертация ишиди ўзбек тилидаги электрон матнлар ишончлигини назорат қилувчи тизимни амалга ошириш масалалри кўрилган. Электрон матнлардан статистик маълумотлар бўйича леммаларни ажратиб олиш тизимни яратишга ёндошув таклиф этилган. Леммалар луғати мавжудлиги сўзлар мофологик таҳлилини соддалаштиради, аниқликни оширади. Леммалар учун таржимани қўшиш машинали таржимонни яратишга хизмат қилади.

Матнлар назорати сифатини таъминлаш ҳамда табиий тилда, ҳусусан ўзбек тилида бериладиган матнлар таркибида хатоларни назорат қилиш ва таҳрирлаш учун маълумтоларга ишлов бериш дастурий тизим учун рационал хотира ҳажмини ҳсиоблаш услубиятлари ишлаб чиқилган.

Ўзбек тилидаги сўз шаклларни таҳлил қилишда формал тавсифлаш қоидалари тадқиқ қилинган ва улар асосида морфологик таҳлил модели ҳамда фойдаланувчидан тилни чуқур билишни талаб этмайдиган, катта бўлмаган ҳажмдаги лўғат билан назоратни амалга ошириш имкониятини берадиган, хатоларни назорат қилиш ва таҳрирлаш дастурий тизимни яратишнинг умумлашган алгоритми ва тузилиши ишлаб чиқилган.

Электрон матнларни назорат қилиш тизимини жорий этиш учун дастурий муҳитни танлаш асосланган. VB.NET дастурлаш тилининг асосий имкониятлари таҳлил қилинган. VB.NETда тақдим этилган матн, тузилмалар билан ишлаш кенг имкониятларидан фойдаланилди ва бу тизимнинг умумий самарадорлигини салмоқли даражада яхшилади.

Морфологик луғат маълумотлар базасини лойиҳалашнинг дастурий модуллари ишлаб чиқилган ва амалга оширилган. Тугалланган мантиқий мазмунга эга бўлган, керакли қайта ишлаш амалларини бажарадиган ва натижавий маълумотларнинг ягоналаштирилган тўпламини ечим сифатида кўрсатадиган анализатор тузилмаси амалга оширилган.

Машинали таҳлил тизимларнинг дастурий таъминотининг вазифалари ва ишга тушириш талаблари аниқланган. Дастлабки маълумотларнинг турли мисолларида морфологик лексиконнинг қайта ишлаш натижалари олинган.

Ҳусусий натижалар сифатида қуйидагиларни кўрсатиш мумкин:

- ўзбек тилига қўлланиладиган морфологик таҳлил усуллари ишлаб чиқилган;
- от ва сифат сўз туркумлари учун сўзнинг негизини башорат қилувчи услублар ишлаб чиқилган;
- амалга ошириш алгоритмлари ишлаб чиқилган ва VB.NET дастурлаш тилида дастурлаш бажарилган;
- маълумотларни узатиш ва уларга ишлов бериш талабларишa жавоб берадиган тизим унумдорлиги таҳлили бажарилган.

Магистратура талабаси И.Химматов

Илмий раҳбар проф. А.Р.Ахатов

MINISTRY OF HIGHER AND SECONDARY SPECIAL EDUCATION OF THE REPUBLIC OF UZBEKISTAN

SAMARKAND STATE UNIVERSITY

Faculty: Mechanics-mathematics	Student of Magistracy: Himmatov Ibodila
Department: Information technologies	The scientific chief: prof.assis. A.R.Akhatov
Academic year: 2015-2016	Speciality: 5A110701 – Information technologies in education

ANNOTATION of MAGISTR'S DISSERTATION

Urgency of theme. At input, transfer and processing of information on COMPUTER can arise of a various sort of mistake in the texts. The reasons of mistakes can be: devices of scanning recognizing program systems, man – operator, processing this information; various handicaps influencing the channel of communication; failures and refusals of electronic tools. The importance of problem of control of information reliability is emphasized by that, what all business information in our republic is conducted in the Uzbek language, however, not all citizens of republic have skill of spelling of the Uzbek texts, therefore in the texts there are additional spelling mistakes. Hence, urgent tasks are development of methods, models, algorithms and program system of processing of the texts in the Uzbek language for the control and correction of mistakes, and also its practical realization in system of electronic document circulation in enterprises and organizations.

The aim of work. Aim is development of algorithms and software to control text information on the basis of models for morphological dictionary management demonstrating an opportunity of effective decision of the task of morphological analysis, as key stage of electronic texts reliability maintenance.

The research problems: Study of linguistic strategy and rules adequate to wordforming laws of language. Search of cases to applicating described grammatic designs for texts. Development morphological analysis ways without the dictionary and technique of analysis wordforms in morphological model. Development algorithm for morphological analysis on a basis of indexing and designing the dictionary morphology. Substantiation of complexity machine analysis for Uzbek sentences, development of a posmorphological analysis technique. Development of general circuit for segment and intrasegment analysis actions. Construction of architecture and software-realized modules for the monitoring system and correction of spelling mistakes on the basis of morphological dictionary management models.

Object of research – texts and documents in electronic documents interchange system of enterprises and organizations**. Subject of research -** methods, algorithms and system for monitoring of text information reliability.

Technique and methods of research: bases of control systems construction, algorithmization theory, theory of transfer and processing of information, theory of programming, Uzbek lexicologica.

The scientific novelty of job consist in offered algorithms allowed to create the module of morphological analysis and capable to realize developed methods for prediction a basis of word and part of speech, self-training analyzer in monitoring system of spelling mistakes for electronic texts in Uzbek language.

The scientific and practical importance of results of research. Results of dissertation can be applied at creation monitoring systems and correction spelling of the Uzbek language, generation and synthesis of the text from speech, machine translation of texts in the Uzbek

language. Also system is possible for using at training to bases of the Uzbek language grammar.

The methods and program system to control of spelling mistakes, developed on the basis of multilevel model to representation of wordforms of Uzbek language do not require deep knowledge of language and solve tasks at small volume of dictionary, work with optimum speed, consume acceptable quantity(amount) of memory.

Structure and volume of job: the dissertation consists of introduction three chapters, conclusion, list of the used literature and applications. The job is stated on 70 pages of the basic text, is illustrated with 13 figures. 59 sources of literature are used.

The basic results of job.

Dissertation researches are considered questions of realization of algorithms for control of electronic texts reliability in the Uzbek language. The approach is offered to creation the system for lemm allocation from electronic texts on the basis of statistical data. The presence of dictionary lemm simplifies morphological analysis of words and raises accuracy. By everyone lemm in the dictionary it is possible addition of translation, it will allow to construct the machine interpreter.

The techniques are developed to calculate the redundancy volume ensuring required quality of texts control and to define the rational volume of memory of information processing system for control and correction mistakes in texts on natural languages, in particular, on Uzbek.

The rules of description the formal analysis of Uzbek wordforms are investigated, on the basis of this the model of morphological analysis, a generalized algorithm to construction and the structure of system for mistakes control and correction are developed not requiring deep knowledge in language of user and allowing to spend control of mistakes with limited volume of the wordforms dictionary.

The choice of program environment is proved for realization the system of electronic texts reliability monitoring. The basic opportunities of the programming language VB.NET are analysed. The wide opportunities of VB.NET during job with the text and structures are used to allow appreciably increase general productivity of system.

Program modules of designing the database of morphological dictionary are developed and realized. The structure of analyzer is realized and it has completed logic sense, carries out necessary processing and returns unified set of resulting data.

Functions and field-performance requirements of the software of machine analysis systems are determined. Results of processing morphological lexicon are received on various examples of initial data.

Private results of researches and practical development are the following:

- methods of morphological analysis are developed with reference to the Uzbek language;
- methods of prediction of a word basis and part of speech for nouns and verbs are created;
- algorithms of realization are developed and programming of system in language VB.NET is carry out;
- analysis of system productivity is carried out, and its results match to requirements of quality for information transfer and processing.

Scientific chief prof. A.R.Akhatov

Student of Magistracy: I.Ximmatov

СОДЕРЖАНИЕ

ВВЕДЕНИЕ …….....3

ГЛАВА I. ОСНОВНЫЕ ПОДХОДЫ К ПОСТРОЕНИЮ МЕТОДОВ КОНТРОЛЯ ДОСТОВЕРНОСТИ ЭЛЕКТРОННЫХ ТЕКСТОВ.....7

1.1. Базовые принципы построения системы контроля достоверности информации ……..….....7

1.2. Определение критериев оценки качественных показателей эффективности контроля достоверности информации 14

1.3. Разработка методики оценивания требуемого рационального объема памяти в системах обработки электронных текстов ..….....21

Выводы к главе 1 …….25

ГЛАВА II. РАЗРАБОТКА АЛГОРИТМА КОНТРОЛЯ ДОСТОВЕРНОСТИ ЭЛЕКТРОННЫХ ТЕКСТОВ НА ОСНОВЕ МОДЕЛЕЙ УПРАВЛЕНИЯ МОРФОЛОГИЧЕСКИМ СЛОВАРЕМ …….…….....27

2.1. Модели безсловарного морфологического анализа .....…….....27

2.2. Модели морфологического анализа на основе словаря словоформ ...29

2.3. Разработка алгоритма морфологического анализа на основе модели индексирования словаря …….....32

2.4. Проектирование алгоритма морфологического лексикона ……43

Выводы к главе 2…….48

ГЛАВА III. РЕАЛИЗАЦИЯ АЛГОРИТМОВ КОНТРОЛЯ ДОСТОВЕРНОСТИ ТЕКСТОВОЙ ИНФОРМАЦИИ НА ОСНОВЕ УПРАВЛЕНИЯ МОРФОЛОГИЧЕСКИМ СЛОВАРЕМ ..…….....50

3.1. Реализация алгоритмов контроля электронных текстов на узбекском языке ……..…….....50

3.2. Выбор программной среды для реализации системы контроля достоверности электронных текстов ..…….....57

3.3. Разработка и реализация программных модулей проектирования базы данных морфологического словаря …….....61

Выводы к главе 3…….69

ЗАКЛЮЧЕНИЕ …….....70.

ЛИТЕРАТУРА …….....71

ПРИЛОЖЕНИЕ …….....76

ВВЕДЕНИЕ

Актуальность темы. В рамках проблем создания единого информационного пространства Республики Узбекистан приоритетным направлением является разработка прогрессивных средств и методов информатизации на базе современных технологий интеллектуального анализа данных, предназначенных для решения широкого круга научно-технических и практических задач. В этой связи приобретает особое значение разработка информационных технологий и технологий создания систем обработки текстовых данных в системах автоматизации делопроизводства [1-6].

Анализ литературы и информации из глобальной сети по данной проблеме показал, что в настоящее время ведутся работы отдельно по частотному и морфологическому анализу, но готовых морфоанализаторов пока нет. Большинство существующих программных продуктов считают словом набор символов “от пробела до пробела”. Для таких программ слова “математика” и “математики” являются совершенно разными, не говоря уж о слове “математический”. Программ, которые бы полностью учитывали особенности узбекского языка и его морфологию, не существует.

Попытки написать морфоанализатор предпринимались и ранее, но в итоге получались программы, неадаптированные для естественных языков. На данный момент разработано небольшое количество частотных анализаторов для английского языка, учитывающих парадигматические изменения. Программы для узбекского языка находятся еще в стадии разработки.

В связи с этим, разработка погрaммммных средств морфологического анализа текстов на узбекском языке представляет собой акутальную тему исследований для решения проблемы обеспечения достоверности электронных документов.

Степень изученности. Среди спектра задач, которые решаются программой автоматизации процессов морфологического анализа, можно выделить следующие основные направления:

1. *Интеллектуальные поисковые системы*. Как известно, нынешнее состояние

поисковых систем оставляет желать лучшего. По запросу пользователя часто выдаётся слишком много информационного шума, либо, наоборот, результаты поиска слишком однонаправлены. Для решения данной проблемы идут активные разработки интеллектуальных поисковых систем, которые будут учитывать по запросу пользователя не только введённое слово, но и смежную информацию.

2. *Машинный перевод*. На данный момент не существует программ, корректно переводящих узбекский текст. Морфологический анализатор позволяет улучшить качество автоматического перевода, поскольку дает возможность согласовывать падежи, числа, склонения переводимых слов. Кроме того, зная морфологическую структуру слова и учитывая статистику частоты его употребления в сочетании с другими словами, упрощается задача подбора синонимов и согласования их в тексте.
3. *Авторизация текста*. В наши дни эта проблема является очень актуальной, и примером тому служит немало судебных процессов, касающихся авторства того или иного текста – большей частью это относится к информации из глобальной сети. Чтобы выяснить, кому принадлежит исходный текст, приходится обращаться к профессиональным лингвистам, которые вынуждены вручную анализировать тексты, выявляя стилистку разных авторов. В ходе работы статистически исследуется использование автором различных междометий, частиц, средней длины слов и т. д.
4. *Составление конкордансов* – словарей, содержащих слова из всех произведений одного автора. Достаточно рутинная работа – проанализировать стилистику какого-либо автора по его произведениям. Благодаря автоматическому разбиению слов на леммы, появляется возможность автоматизированного анализа авторских текстов.

Поставленная в настоящей работе тема исследований по разработке методов, моделей, алгоритмов для построения системы обеспечения достоверности электронных текстов на основе на основе управления морфологическим словарем является малоизученной и требует специального

подхода для устранения существующих недостатков в известных методах и обеспечения требуемой достоверности информации при незначительных временных и материальных затратах.

Целью настоящей работы является разработка алгоритмов и программных средств контроля текстовой информации на базе моделей управления морфологическим словарем, демонстрирующих возможность эффективного решения задачи морфологического анализа, как ключевого этапа обеспечения достоверности электронных текстов.

Задачи исследований. В соответствии с целью поставлены и решены следующие теоретические и практические задачи:

1. Изучение лингвистических стратегий и правил, отвечающих словообразовательным законам языка. Поиск случаев применения описываемых грамматических конструкций в корпусе текстов.
2. Разработка способов морфологического анализа без словаря и методики анализа словоформ в морфологической модели.
3. Разработка алгоритма морфологического анализа на основе индексирования и проектирование словарной морфологии.
4. Обоснование сложности машинного анализа узбекского предложения, разработка методики постморфологического анализа.
5. Разработка общей схемы действий сегментационного анализа и внутрисегментного анализа
6. Построение архитектуры и программно-реализуемых модулей системы контроля и исправления орфографических ошибок на основе моделей управления морфологическим словарем.

Предметом исследования является структура электронных текстов на узбекском языке, законы построения узбекских слов и предложений.

Методы исследования:

- Создание и пополнение лексиконов, содержащих необходимую для анализа морфологическую и грамматическую информацию;
- Создание динамических структур данных для представления и хранения

синтаксической информации и программное моделирование процесса анализа на вычислительных устройствах;

- Создание отладочного массива слов и словосочетаний, охватывающего все множество отраженных в модели грамматических явлений, тестирование системы в пространстве реальных текстов.

. **Научная новизна работы** состоит в том, что предложенный алгоритм позволяет создать модуль морфологического анализа и реализовать разработанные методы предсказания основы слова и части речи, а также самообучающего анализатора в системе контроля орфографических ошибок для электронных текстов на узбекском языке.

Научная и практическая значимость результатов исследования. Результаты диссертационной работы могут быть применены при создании систем контроля и коррекции орфографии узбекского языка, генерации и синтеза текста из речи, машинного перевода текстов на узбекском языке. Также систему возможно использовать при обучении основам грамматики узбекского языка.

Апробация работы. Основные выводы и научные результаты диссертационной работы доложены на научно-практических конференциях (2014-2016 г.г.). По теме диссертации опубликованы 3 тезисв докладов

Структура и объем работы: Диссертация состоит из введения, трёх глав, заключения, списка литературы из 50 наименований и приложения. Общий объем работы - 88 страниц, основной текст – 66 страниц.

В диссертационной работе использованы опубликованные научные работы проф. И.И.Жуманова, доц. А.Р.Ахатова. Пользуясь случаем, автор считает возможным поблагодарить научного руководителя за консультации и уделенное внимание во время подготовки диссертации.

ГЛАВА I. ОСНОВНЫЕ ПОДХОДЫ К ПОСТРОЕНИЮ МЕТОДОВ КОНТРОЛЯ ДОСТОВЕРНОСТИ ЭЛЕКТРОННЫХ ТЕКСТОВ

Целью настоящей главы является определение основных подходов, принципов и методов построения системы обработки информации для контроля и коррекции ошибок в электронных текстах на естественных языках, а также разработка методик оценки и анализа вероятностных и количественных показателей эффективности исследуемой системы. В соответствии с целью главы ставятся следующие теоретические и практические задачи:

- выбор направлений проводимых исследований, определение базовых принципов построения компьютерной системы контроля и коррекции ошибок, анализ вероятности искажений на этапах ввода, передачи и обработки электронных текстов;

- разработка методики определения объема избыточности, который обеспечивал бы требуемую достоверность информации при применении методов ипрограммных средств контроля и коррекции ошибок, основанных на использовании избыточностей различной природы,

- разработка методик синтеза вероятностных процессов, определения количества информации: с учетом происходящих вероятностных процессов при обычным приеме сообщения и при применении корректирующего кода.

- разработка методики определения требуемого рационального объема памяти компьютерной системы обработки информации с встроенными средствами контроля и коррекции ошибок в электронных текстах.

1.1. Базовые принципы построения системы контроля достоверности информации

В настоящем параграфе излагаются основные подходы к созданию методов, алгоритмов контроля и коррекции ошибок в текстах при обработке данных в системах электронного документооборота (СЭД) предприятий.

Одним из важных критериев функционирования СЭД предприятий является достоверный обмен данных. Однако, в реальных условиях, достоверность информации очень низка и равна примерно $3{,}4 \cdot 10^{-2}$ ош/знак. Установлено, что

около 85% ошибок в общем объеме искажений принадлежат человеку-оператору, процессам сканирования и распознавания. Причем для нормального функционирования системы требуется повысить достоверность обрабатываемой информации до 10^{-5}-10^{-6} ош/знак, что подчеркивает актуальность решения проблемы построения программной системы контроля (обнаружения) и коррекции (автоматического исправления) ошибок в текстах [19].

Изложим предлагаемые нами основные подходы к созданию системы контроля и коррекции ошибок в текстах, которые отражаются моделью, приведенной на рис.1.1.

Способы ввода и передачи информации. Текстовая информация в виде машинописных текстов, документов, графиков и т.д. может вводиться человеком-оператором, посредством сканирующих устройств, включая распознающие программные комплексы, а также передаваться в виде файлов машинными носителями, по электронной почте и др. Этапам передачи и обработки текстов, где происходит процесс контроля и коррекции ошибок, как правило, предшествуют кодирование и декодирование текстовой информации. В связи с этим, в плане данной работы, наряду с другими, нами также решаются задачи, связанные с разработкой методов, алгоритмов и программ контроля и коррекции ошибок на основе эффективных методов кодирования, сжатия и декодирования информации.

Кодирование информации. Входной текст, представляемый для решения задач управления и обработки информации обычно кодируется. Для кодирования текстовой информации можно использовать различные способы, в частности коды Шеннона-Фано, Хаффмана, арифметического кодирования; словарные методы: Зива-Лемпеля, Лемпеля-Зива-Велча и алгоритмы, использующего преобразование Барроуза-Вилера, машинный код ASCII и др. Доказано, что из-за простоты и большой эффективности в создаваемой ИС можно отдавать предпочтение использованию арифметического кода [18,20].

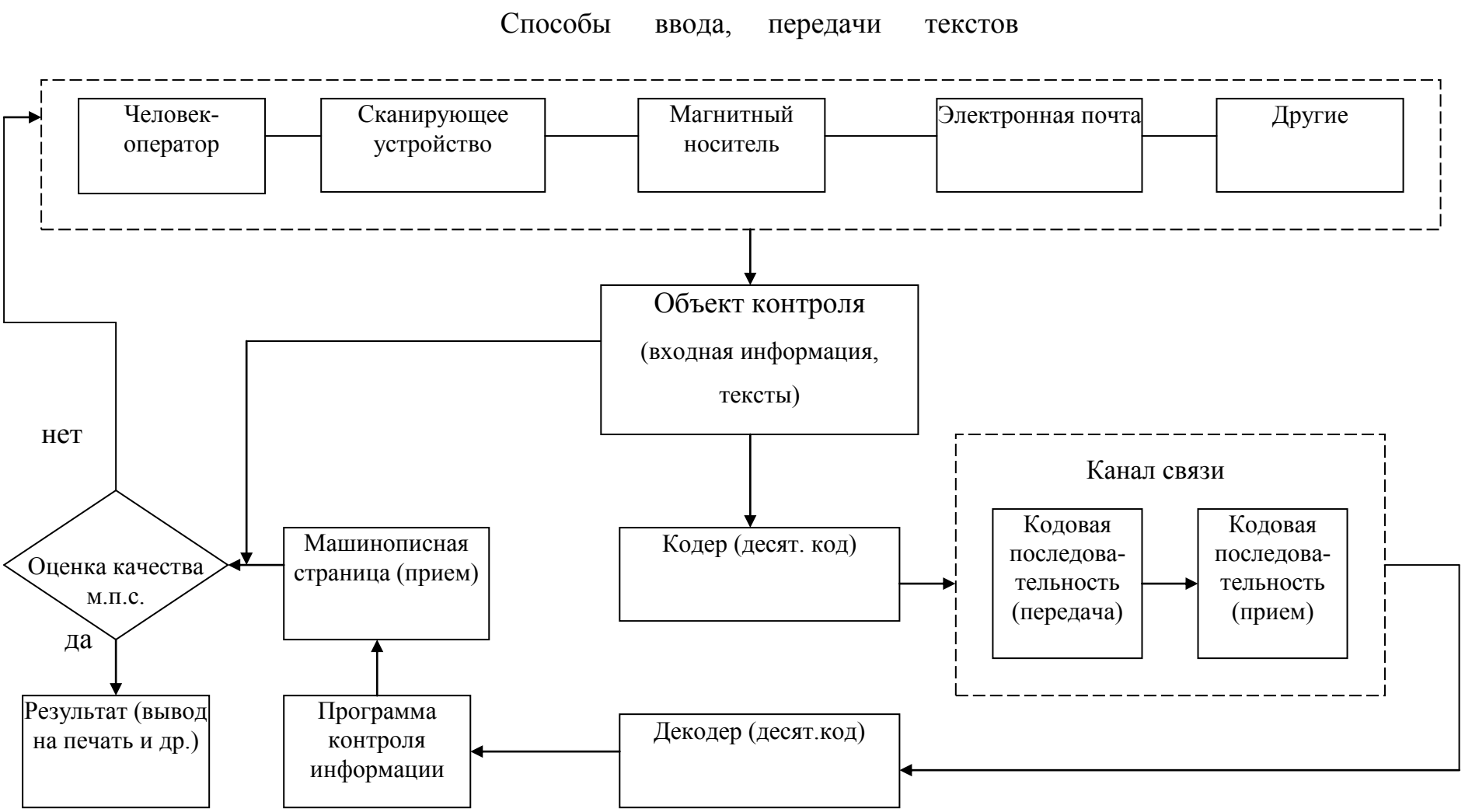

Рис. 1.1. Модель функцинирования системы контроля и коррекции ошибок в текстах

Особенность данного метода заключается в том, что передаваемый текст, как правило, кодируется вещественными числами или же можно использовать десятичные числа. Это дает возможность построить алгоритм адаптации границ для декодирования, который можно использовать для контроля достоверности реальных декодируемых текстов. В связи с этим, для контроля и коррекции ошибок в текстах поставлена и решается задача, связанная с разработкой, исследованием и применением программных методов, основанных на использовании особенностей, схемы и правил рекомендуемого арифметического кодирования.

Программные методы контроля информации. В зарубежной и отечественной практике программные методы контроля, обнаружения и автоматического исправления ошибок в текстовой информации изучены недостаточно и пока отсутствуют разработки, эффективно применяемые на практике. В [9,10] результатами исследований доказывается возможность применения программных методов контроля цифровой информации для обнаружения одиночных и двукратных орфографических ошибок в текстах, например, за счет использования алгоритмов линейного, модульного и правил плоскостного суммирования.

Выше было отмечено, что ошибки человека-оператора, сканирования и распознавания составляют значительную долю в общем объеме искажений. Вместе с тем такие ошибки характеризуются как ошибки большой кратности (k-грамм). Следовательно, задачи контроля и коррекции ошибок в текстах должны решаться в новой постановке, где будут учитываться отмеченные условия переработки информации. Причем требуется, чтобы встраиваемая в компьютерную систему обработки данных программная система обеспечивала комфортные условия при обнаружении и исправлении ошибок на основе использования современных компьютерных технологий.

Наряду с применением программных методов, использующих искусственную избыточность эффективно также использование естественной избыточности для контроля и исправления ошибок в текстах [23-25].

При постановке задачи контроля информации на основе естественной избыточности, программный комплекс коррекции ошибок может быть реализован на основе разработки следующих методов контроля ошибок: по границам кодовых множеств, перекодированной текстовой информации; по специальным справочникам словоформ естественного языка. Помимо этих можно использовать методы, учитывающие логические связи между последовательностями фраз, слов, букв или специфику системы кодирования; контроль по границам допустимых кодов и др.; методы учитывающие статистические связи и корреляции данных; семантические методы, учитывающие свойства языка и структуру образования словоформ; методы обнаружения грамматических ошибок, основанные на морфологическом анализе; неморфологические методы (словарные и бессловарные). К словарным относятся методы, опирающиеся на использование неструктурированного списка всех допустимых словоформ, а к бессловарным – методы, подвергающие проверке часть словоформ (методы диграмм, триграмм, n-грамм) и метод хеш-кодов; методы, на основе применения алгоритмов классификации и распознавания текстов исследуемого языка и др. [24-27,32-34].

Следует отметить, что среди указанных методов в качестве основного для проверки орфографических ошибок в текстах можно выделить метод морфологического анализа словоформ. В связи с этим в рамках настоящей работы планируется проведение подробных исследований, связанных с разработкой программной системы обработки данных для контроля орфографии узбекского языка (латиница, кириллица).

Как известно, естественные языки в соответствии с принципами своего строения делятся на три группы: аналитические, флективные и агглютинативные. Агглютинативные языки, к которым относятся большинство тюркских, в частности, узбекский, характеризуются промежуточным положением между аналитическими и флективными языками [7,8,38,39,41,53]. С одной стороны, у них сохраняется весьма богатая система словоизменительных и словопорождающих аффиксов, но, с другой стороны, эта система характеризуется значительной конструктивностью и простотой. Однако, несмотря на

относительную простоту и конструктивность тюркских языков, проблемами разработки spell-checker‘ов для них лингвисты занимаются слабо, сосредоточившись в основном, на европейских языках. Нам известен список научных коллективов, занимающихся разработками автоматизированной обработки речи и текстов на естественных языках [11-13, 18-21, 31-36, 51-52].

Наиболее продвинутые исследования в этой области проводятся научной группой Кемаля Офлазера в Турции в рамках проекта НАТО.

Проблеме разделения потока текста на составные части посвящены многочисленные работы американских лингвистов. Фирмой Xerox предложены средства для автоматизированной обработки текстов на естественных языках, основанные на применении многоуровневого описания грамматики. Описание лексики узбекского языка с помощью предложенной американскими лингвистами двухуровневой грамматики является правомерной задачей в рамках построения системы орфографического контроля современного литературного узбекского языка [47-50].

Следующей проблемой является учет современного состояния узбекской орфографии – одновременное присутствие в ней двух различных систем графики (кириллицы и латиницы). Узбекский spell-checker должен обладать способностью проверять орфографию в обоих представлениях и при необходимости переводить из одного представления в другое. Наиболее сложной и трудоемкой задачей является подготовка достаточно представительного словаря корней слов современного узбекского литературного языка. Здесь большую помощь оказывают выпущенные недавно АН Узбекистана орфографические словари на латинице.

Таким образом, исследование основных аспектов теоретических и практических задач, на результатах которых может быть построена компьютерная система, основанная на морфологическом анализе текстов на узбекском языке является интересной и нерешенной проблемой.

Оценка качества текстов. Наряду с вышеуказанными, ставится задача связанная с проверкой качества текста, решение которой будет требовать

разработку специфичных алгоритмов и программ для сравнения исходного и проконтролированного текста. Можно предложить несколько вариантов решения этой задачи: побуквенное, символьное сравнение; сравнение слов; сравнение слов с выделением корней; сравнение строк; сравнение частотных характеристик слов, массивов, строк и страницы; сравнение интервалов появления букв, символов, слов, строк в странице.

Таким образом, в настоящем параграфе предложены основные подходы к созданию компьютерной системы контроля и коррекции ошибок в текстах, сформулированы задачи, определяющие следующие направления теоретических и практических исследований:

разработка методик анализа вероятностных процессов, происходящих в различных условиях передачи и обработки информации и определения оптимального объема словаря словоформ узбекского языка;

разработка методов и алгоритмов контроля и коррекции текстов на основе: алгоритмов оптического распознавания текстов; вероятностной модели совершения ошибок; метода арифметического кодирования; программных методов контроля информации по линейным, модульным, плоскостному суммированию и методов морфологического анализа на основе многоуровневой модели представления словоформ узбекского языка.

1.2. Определение критериев оценки качественных показателей эффективности контроля достоверности информации

Принципиальной основой построения программной системы обработки текстов для повышения достоверности информации является использование искусственной и естественной избыточности. Искусственная избыточность образуется путем ввода в структуру передаваемого сообщения дополнительной контролирующей информации, например, вводимых проверочных символов в кодовую последовательность, либо контрольных сумм при использовании программных методов контроля ошибок. Естественная же избыточность, как правило, содержится в составе оригинала обрабатываемого текста или в перекодированном его варианте и обусловливается корреляцией, либо неравномерными распределениями вероятностей появления букв и символов в строках или страницах [18-20].

Увеличение объема избыточности информации с одной стороны обеспечивает большую достоверность информации, однако с другой стороны это приводит к повышению затрат на ее передачу и обработку.

В связи с этим возникает задача, связанная с разработкой методики определения такого объема избыточности, который обеспечивал бы требуемую достоверность информации при наименьших затратах.

Решение этой задачи состоит из нескольких этапов. Первым из них является синтез вероятностных процессов, происходящих при контроле и передаче информации; второй – определение количества информации с учетом происходящих вероятностных процессов; определение количества информации при обычным приеме сообщения; определение количества информации при применении корректирующего кода и определение требуемого объема избыточности для рассмотренных условий передачи информации.

Синтез вероятностных процессов. Обозначим через P_{α_i} среднюю вероятность ошибок при передаче α_i-го символа (десятичного знака, кодовой

комбинации), а через $P_{\alpha_{i\,j}}$ - условную вероятность приема СВ α_j при передаче СВ α_i или назовем ее вероятностью перехода α_i в α_j $(\alpha_i \to \alpha_j)$:

а) $P_{\alpha_{i\,j}} = 1$, при $j = i$;

б) $P_{\alpha_{i\,j}} = 0$, при $j \neq i.$;

в) $P_{\alpha_{i\,j}} = \frac{1}{B}$, при $0 < \alpha_j < B$.

Здесь B – диапазон изменения СВ α_i, который определяется соотношением $B=X^m$, где X – основание кода; m – длина кодовой последовательности.

Условие (а) означает безошибочную передачу информации, (б) – передачу информации с полным искажением, а условие (в) означает, что при передаче знака α_i будет принят любой знак α_j с постоянной условной вероятностью, равной B^{-1}. Значение СВ α_i могут задаваться в нескольких разрядах (единичных, десятичных, сотых и т.д.), что представляет *m*-последовательность десятичных кодов. Объединение таких *m*-последовательностей образует длину обрабатываемых сообщений.

Обозначим вероятность приема одного кодового сообщения как P (S). С введением процедуры контроля информации вероятность P (S) распадается на две части: вероятности правильного приема сообщений – Q(S), и вероятности неправильного приема – $P_Н(S)$, т.е. $P(S) = Q(S) + P_Н(S)$.

При идеальной процедуре контроля информации $P(S) = Q(S)$ При использовании рационального метода контроля информации- $P_{ост}(S) = 1 - Q(S) \to \min$. При неэффективной процедуре контроля имеем $P(S) = P_Н(S)$.

Возникает вопрос, как минимизировать вероятность необнаружения ошибок до допустимых пределов достоверности передачи и обработки информации, т.е. значения остаточной вероятности ошибок должно быть на три-четыре порядка ниже существующего значения средней вероятности ошибок ввода, передачи и обработки текстовой информации, т.е.

$$P_{ост}(S) << P_{\alpha_i}, \text{ где } P_{\alpha_i} = 10^{-3} \div 10^{-4}.$$

Выполнение такого условия требует использования избыточности информации. Пусть будет принято решение, что передавалось кодовое слово S_0.

Тогда вероятность $Q(S_0/S_1')$ того, что принято правильное решение определяется соотношением $Q(S_0/S_1') = P(S_0/S_1')$, где $P(S_0/S_1')$ – условная вероятность того, что при приеме S_1' передавалось кодовое слово S_0.

Вероятность ошибочного решения имеет вид

$$P(S_0/S_1') = \sum_{i=1}^{2^m-1} (1 - P_0(S))^{-1} P(S_i/S_i'),$$

где $P_0(S)$ – вероятность того, что в кодовом слове будет обнаружена ошибка. Для всех $i \neq 1$ $P(S_i/S_1') = P(S_0/S_1')$, а для $i = 1$, например, для семизначной двоичной последовательности

$$P(S_i/S_1') = (1-P)^7/(1-P)^7 + 7P^4(1-P)^3,$$

где P – вероятность искажения одного двоичного символа в коде.

Тогда $P(S_0/S_1') = (1-P)^7 + 6P^4(1-P)^3/(1-P)^7 + 7P^4(1-P)^3 \approx 1$.

Вероятность правильного решения $Q(S_0/S_1') \approx 7P^4 << 1$.

Обычно $P << 1$, поэтому $Q(S_0/S_1') << P(S_0, S_1')$.

Если будем считать, что в принятой последовательности S_1' соответствует переданная S_1, то

$$Q(S_1/S_1') = [1 - P_0(S)]^{-1}(1-P)^7 \approx 1.$$

Вероятность ошибочного решения

$$P(S_1/S_1') = (2^3 - 1)[1 - P_0(S)]^{-1}.$$

Для рассматриваемого случая она равна $P^4(1-P)^3 \approx 7P^4 << 1$, следовательно

$$Q(S_1/S_1') >> Q(S_i/S_i') \quad \text{для всех } i \neq I..$$

Значит

$$Q(S_j/S_j') >> Q(S_i/S_j') \text{ для всех } i \neq j.$$

Вероятность Q(S) определяется как среднее значение вероятности $Q(S_i)$ правильного приема кодового слова S_i

$$Q(S) = \sum_{i=1}^{2^m-1} P(S_i)(1-P)^m = (1-P)^m.$$

Следует заметить, что $P(S_i) = 2^{-m}$ при $B = 2^m$

$$P_H(S) = \sum_{i=0}^{2^m-1} P(S_i) \sum_{\substack{j=0 \\ j\neq i}}^{2^m-1} P(P_j) = \sum_{j=0}^{2^m-1} P(R_j),$$

где R_j совпадают с ненулевыми элементами кодового множества.

И наконец,

$$P_0(S) = \sum_{j=0}^{2^m-1} P(R_j) = 1-(1-P)^m - \sum_{j=1}^{2^m-1} P(R_j).$$

Определение количества информации. Теперь ставится задача, связанная с определением количества информации, достаточной для обнаружения ошибок в текстовых сообщениях.

Известно, что количество принятой в сообщении информации измеряется изменением меры неопределенности сообщения до и после приема. Мерой априорной неопределенности сообщения является энтропия алфавита сообщений. При передаче формализованной информации все сообщения равновероятны и $H(S) = -\log 2^m = m$.

Остаточная (апостериорная) энтропия после декодирования и выдачи знака определяется средней энтропией по алфавиту принимаемых сообщений при условии, что известен принятый знак $S_k^{/}$:

$$H(S_i/S_k^{/}) = -\sum_{k=1}^{2^m} P(S_k^{/}) \sum_{i=1}^{2^m} P(S_i/S_k^{/}) \log P(S_i/S_k^{/}).$$

Отсюда среднее количество информации о переданном сообщении, содержащееся в принятом знаке, определяется как

$$J(S_i/S_k^{/}) = H(S_i) - H(S_i/S_k^{/}) = -\sum_{i=1}^{2^m} P(S_i) \log P(S_i) + \sum_{k=1}^{2^m} P(S_k^{/}) \sum_{i=1}^{2^m} P(S_i/S_k^{/}) \log P(S_i/S_k^{/}).$$

При декодировании с обнаружением ошибок для введенных выше ограничений на источники информации и корректирующего кода количество информации определяется из выражения

$$J(S_i/S_k^{/}) = m + \frac{1}{2^m} \sum_{k=1}^{2^m} \sum_{i=1}^{2^m} P(S_i/S_k^{/}) \times \log(S_i/S_k^{/}) = m + \sum_{i=1}^{2^m} P(S_i/S_k^{/}) \log(S_i/S_k^{/}) =$$

$$= m + Q^{/}(S) \log Q^{/}(S) + \sum_{i=1}^{2^m} P_{ki}^{/}(S) \log P_{ki}^{/}(S).$$

Чтобы доказать достоверность полученного результата выполним следующие преобразования:

$$\sum_{i=1}^{2^m} P(S_i/S_k')\log P(S_i/S_k') = (1-P_0(S))^{-1}\times P(S_0/S_1')\log(1-P_0(S))^{-1}P(S_0/S_1') +$$

$$+\sum_{i=1}^{2^m-1} P(S_i/S_k')\log P(S_i/S_k') = (1-P_0(S))^{-1}\times(1-P)^n\log(1-P_0(S))^{-1}(1-P)^n +$$

$$+\sum_{i=1}^{2^m-1} P(S)\times\sum_{j=0}^{2^m-1} P(P_j)\log\sum_{i=1}^{2^m-1} P(S_i)\sum_{j=0}^{2^m-1} P(P_j) = (1-P_0(S))^{-1}Q\log(1-P_0(S))^{-1}Q +$$

$$+\sum_{i=1}^{2^m-1} P(S_i)\times\sum_{\substack{j=0\\ j=i}}^{2^m-1} P(R_j)\log\sum_{i=1}^{2^m-1} P(S_i) + \sum_{i=1}^{2^m-1} P(S_i)\times\sum_{\substack{j=0\\ j=i}}^{2^m-1} P(R_j)\log\sum_{\substack{j=0\\ j=i}}^{2^m-1} P(R_j) =$$

$$= (1-P_0(S))^{-1}Q\log(1-P_0(S))^{-1}Q + P_H(S)\log\sum_{i=1}^{2^m-1} P(S_i) + P_H(S)\log\sum_{\substack{j=0\\ j\neq i}}^{2^m-1} P(R_j) =$$

$$= (1-P_0(S))^{-1}Q\log(1-P_0(S))^{-1}Q + (1-P_0(S))^{-1}P_H(S)\log\sum_{\substack{j=0\\ j\neq i}}^{2^m-1} P(R_j).$$

Здесь $(1-P_0(S))P_H(S)\log\sum_{i=1}^{2^m} P(S_i) = 0$, так как $\sum_{i=1}^{2^m} P(S_i) = 1$, и приведем к виду

$$J(S_i/S_k')m + Q'\log Q' + P_H'\log P_H(S),$$

где $Q'(S) = Q'(S) = Q(S)[1-P_0(S)]^{-1}$; и $P_H'(S) = P_H(S)[1-P_0(S)]^{-1}$.

Окончательное выражение приведем к следующему виду:

$$J(S_i/S_k') = m + \frac{Q(S)}{1-P_0(S)}\log Q(S) - \frac{Q(S)}{1-P_0(S)}\log[1-P_0(S)] + \frac{P_H(S)}{1-P_0(S)}\times$$

$$\times\log P_H(S) - \frac{P_H(S)}{1-P_0(S)}\log[1-P_0(S)] - \frac{P_H(S)}{1-P_0(S)}\log(2^m-1) = m - \frac{P_H(S)}{1-P_0(S)}\times$$

$$\times\log(2^m-1) - \log[1-P_0(S)] + \frac{Q(S)}{1-P_0(S)}\times\log Q(S) + \frac{P_H(S)}{1-P_0(S)}\log P_H(S)$$

При вероятности $P_H(S) = 0$, то $J(S_i/S_k') = m$, поскольку $Q(S) = 1-P_0(S)$. При другом граничном условии $Q(S) = 0$.

Так как при этом $P_H(S) = 1-P_0(S)$, то получим

$$J(S_i/S_k') = m - \log_2(2^m-1).$$

В данном случае $J(S_i/S_k')$ определяет тот минимум дополнительной информации о сообщении, который необходим для устранения потери из-за неправильного приема.

Определение количества информации при обычном приеме. Потеря информации в результате необнаруженных ошибок в информации зависит в

первую очередь от эффективности применяемых методов контроля. Рассмотрим обычный прием информации без применения процедур контроля ошибок.

Вероятность неправильного приема кодовой комбинации S_i определяется как

$$P_H(S) = 1 - (1-P)^m \approx mP,$$

где $Q(S) = (1-P)^m$-вероятность правильного приема.

Тогда остаточная вероятность

$$P_0(S) = 1 - Q(S) - P_H(S).$$

В рассматриваемом случае $P_0(S) = 0$, тогда

$$J(S_i / S_k^{/}) = m - \frac{mP}{1-P_0(S)}\log(2^m - 1) - \log[1-P_0(S)] + \frac{(1-P)^m}{1-P_0(S)}\log(1-P)^m +$$

$$+ \frac{mP}{1-P_0(S)}\log(mP) = m - mP\log(2^m - 1) - 0 + (1-P)^m \log(1-P)^m + mP\log mP =$$

$$= m - mP[\log(2^m - 1) - \log mP] + (1-P)\log(1-P)^m - 1 = (m-0) - mP\log\frac{(2^m-1)}{mP} +$$

$$+ (1-P)^m \times \log(1-P)^m \approx m - mP\log\frac{(2^m-1)}{mP} \approx m(1-P)\log\frac{(2^m-1)}{mP}.$$

Определение количества информации при применении корректирующего кода. В вычислительных сетях для передачи информации используется семизначный код с добавлением трех проверочных символов, т.е. образуют корректирующий код (7,3) В данном случае вероятность правильного приема $Q(S) = (1-P)^m$, где $m = 10$.

Вероятность браковки знака в результате обнаружения ошибок

$$P_0(S) = (1-P)^7 + 7P^4(1-P)^3.$$

Вероятность неправильного приема

$$P_H(S) = (2^m - 1)P^4(1-P)^3.$$

Требуемое количество информации определяется как

$$J(S_i/S_k^{/}) = m\frac{(2^m-1)P^4(1-P)^3}{[1-(1-P)^7+7P^4(1-P)^3}\log(2^m-1) - \log[1-(1-P)^7+7P^4(1-P)^3 +$$

$$+\frac{(1-P)^m}{[1-(1-P)^7+7P^4(1-P)^3}\log(1-P)^m + \log(2^m-1) - \log[1-(1-P)^7+7P^4(1-P)^3 +$$

$$+\frac{(1-P)^m}{[1-(1-P)^7+7P^4(1-P)^3}\log(1-P)^m + \frac{(2^m-1)R^4(1-P)^m}{[1-(1-P)^7+7P^4(1-P)^3}\log(2^m-1)P^4(1-P)^3.$$

Учитывая, что $P << 1$ данное выражение можно переписать в виде:

$$J(S_0/S_k^{/}) = m-(2^m-1)\log(2^m-1) - \log 7P^4(1-P) + (2^m-1)\log[(2^m-1)P^4].$$

Определение требуемого объема избыточности информации. Избыточность передаваемых данных определяется формулой

$$R(S_i) = 1 - \frac{H(S_i)}{H(S_i/S_k^{/}).}$$

В эту формулу подставив выражение максимальной энтропии и условной энтропии $H(S_i/S_k^{/})$.определяем $R(s_i) = 1 - \frac{1}{(1-P)}\log\frac{(2^m-1)}{nP}$.

Изложенную методику нетрудно применить и для различных вариантов контроля информации. При этом достаточно знания только вероятности неправильного приема (необнаруженной ошибки) сообщения.

Нами проведены оценки вероятностей необнаруженных ошибок для различных программных методов контроля достоверности информации.

Установлено, что для методов контроля достоверности информации, основанных на статистических свойствах данных, объем естественной избыточности должен быть не менее 0,4.

1.3. Разработка методики оценивания требуемого рационального объема памяти в системах обработки электронных текстов

Решение задач построения разрабатываемой программной системы обработки информации связано с расчетом рационального объема памяти, требуемого при контроле и коррекции ошибок в электронных текстах. В связи с этим, проведение исследований с целью разработки методики определения требуемого объема словоформ естественных языков, на которые основываются алгоритмы и программные средства, представляет большой теоретический и практический интерес.

В настоящем параграфе поставлена задача и разработаны алгоритм контроля ошибок и методика оценки параметров алгоритма и определения требуемого рационального объема памяти компьютерной системы обработки информации при контроле и коррекции ошибок.

Ставится задача проверки правильности написания или передачи слова на основе словаря словоформ естественного языка, например, узбекского. Однако, решение такой задачи связано с использованием большого объема памяти вычислительных устройств. Следовательно, нужно построить алгоритм, удовлетворяющий временным критериям и потребляющий приемлемое количество памяти. Для решения этой задачи будем считать, что заранее задано множество правильных словоформ. Также принимается допущение о том, что одно из миллиона неправильных слов не будет обнаружено, т.е. достоверность контролируемого слова должна быть примерно $\approx 10^{-6}$. При этих условиях изложим алгоритм контроля ошибок в текстах.

Алгоритм контроля ошибок. Задаемся оператором

$$F(y) = U[F(\alpha)],$$

который выбирает функции $F_1, F_2, \ldots, F_k$, отображающие множество всех слов в множество целых чисел от 0 до N-1, где $F(\alpha)$ – преобразования некоторой функции в функцию $F(y)$.

Эти функции $F_1, F_2, \ldots, F_k$ считаются независимыми и равномерно распределенными случайными величинами. Будем считать, что k измеряется в десятках, N – в миллионах, а функции F_i вычисляются достаточно быстро с помощью встроенных эвристических процедур.

Обозначим множество всех правильно написанных слов буквой G, а количество его элементов - n.

Рассмотрим набор чисел $\{H_0, H_1, \ldots, H_{N-1}\}$, определяемый так: $H_1 = 1$, если находится $g \in G$ и $i \in \{1,2,\ldots,k\}$ такое, что $F_i(g) = 1$; в противном случае $H_1 = 0$.

Параметры k и N, множество слов G и функции F_i задаются заранее. Тогда можно вычислить и записать на диск набор чисел H_i, каждое из которых задается одним битом. Однако, во избежание медленной работы во время выполнения программы, этот набор необходимо разместить в произвольно адресуемую память быстрого доступа.

Контроль правильности написания слова w заключается в проверке выполнения условия

$$H_{\{F_i(w)\}} = 1 \text{ для всех } i \in \{1,2,\ldots,k\}.$$

Если выполняется данное условие, т.е. слово проходит этот тест, то оно считается правильно написанным; иначе оно рассматривается, как неправильное. Заметим, что слово из G всегда будет определено.

Оценки параметров алгоритма. Вводимый алгоритм не обнаруживает ошибки двух родов:

Первого рода – слово w написано неправильно, а алгоритм считает правильным, т.е. w не принадлежит множеству словоформ G;

Второго рода - правильное слово w будет засчитано как неправильное.

Обозначим ошибки первого рода как P_1, а второго рода - P_2 P_2. Назовем вероятность P_1 - ошибкой типа «пропуск», а вероятность P_2 - ложной тревогой.

Ошибки второго рода, то есть типа ложная тревога, характерны для словарных методов. Они связаны с неполнотой словаря и как правило легко устраняются путем внесения неопознанных слов в словарь. Ошибки первого рода,

при которых система пропускает реальные ошибки, характерны для несловарных методов типа *к*-граммного контроля. Такие ошибки являются в гораздо большей степени неприемлемыми для практически используемой системы орфографического контроля. Кроме того, они являются практически неустранимыми. Так, в слове **очепатка* нет ни одной недопустимой или малочастотной триграммы.

Алгоритм считается эффективным, если уменьшает общую вероятность необнаруженных ошибок, т.е.

$$P_1 + P_2 < P,$$

где P – средняя вероятность неправильного написания слова, которая установлена статистикой ошибок.

Теперь оценим, каковы должны быть параметры k и N, то есть, сколько нужно функций и какова должна быть длина используемого массива. Зададимся для этого допустимой вероятностью ошибки второго рода.

В связи с условием задачи о том, что множество словоформ заранее определено и правильно сформировано в памяти вқчислительного устройства, ниже, в решениях задачи, вероятность ошибок первого рода не учитывается.

Установим, сколько единиц в наборе H_i.

Поставим задачу так: есть m кодовых наборов, каждый из которых случайным образом покрывает одно из N слов. Спрашивается, сколько в среднем слов будет накрыто кодами. Это довольно сложная комбинаторная задача. Мы ограничимся некоторыми приемами ее решения.

Зафиксируем N и рассмотрим среднее количество покрытых слов, как функцию $f(m)$, зависящую от m.

Во-первых, ясно, что $f(1)=1$. Получим рекуррентную зависимость $f(m+1)$ от $f(m)$. Число $f(m+1)$ получается следующим образом:

$f(m)$ слов уже покрыто, когда мы отождестляем $(m+1)$-ую кодовую комбинацию;

вероятность того, что она закроет новое слово равна

$$\frac{N - f(m)}{N}.$$

Общее количество покрытых слов равно

$$f(m) + \frac{N - f(m)}{N}.$$

Таким образом

$$f(m+1) = 1 + f(m)(1 - \frac{1}{N}).$$

Решая это рекуррентное уравнение и учитывая то, что $f(1) = 1$ получаем

$$f(m) = N[1 - (1 - \frac{1}{N})^m].$$

Преобразовав это выражение как

$$f(m) = N[1 - (1 - \frac{mN}{m})^m]$$

и используя $\lim_{N \to \infty} (1 + \frac{k}{N})^N = e^k$ из теории пределов получим, что количество функции в наборе определяется как

$$f(m) \approx N(1 - e^{\frac{-m}{N}}).$$

Оценка требуемой памяти. Пусть b - произвольное слово, не принадлежащее G. Для фиксированного i имеем:

$$P(H_{\{F_{i(b)}\}} = 1) \approx \frac{N(1 - e^{\frac{-nk}{N}})}{N} = 1 - e^{\frac{-nk}{N}}.$$

Используя предположение о независимости F_i, представляем вероятность ошибки второго рода, как

$$P_2 \approx (1 - e^{\frac{-nk}{N}})^k.$$

Таким образом, требуется, чтобы выполнялось следующее неравенство:

$$(1 - e^{\frac{-nk}{N}})^k < P,$$

которое можно переписать так:

$$N > -\frac{nk}{\ln(1-P^{\frac{1}{k}})},$$

где величина N – это длина массива, который нужно хранить в памяти во время работы программы.

Теперь ставится задача, связанная с минимизацией потребляемой памяти. Для этого рассмотрим функцию

$$f(x) = -\frac{nx}{\ln(1-P^{\frac{1}{x}})}$$

при фиксированных P и n.

Дифференцируя и решая получающееся уравнение, а также изучая поведение функции при $x \to 0$ и $x \to \infty$, получаем, что ее минимум достигается при $x = -\log_2 p$ и соответствующее значение

$$N \approx -2n(\log_2 p).$$

Отметим, что оптимальное количество функций не зависит от P. А это выявляет, что при примерном количестве словоформ узбекского языка $n = 2^{16}$ целесообразно использовать 16 функций F_i; при $n = 2^{20}$ (примерное количество словоформ русского языка) целесообразно использовать 20 функций F_i, при $n = 2^{14}$ (словоформ английского) используется 14 функций F_i. Для обеспечения $P_2 \le 2^{-20}$ (10^{-6}) нужен массив H_i из $2^5 \ldots 2^{20}$ битов.

Выводы к главе 1

По результатам теоретических и практических исследований, проведенных в данной главе, можно заключить следующее:

1. Определены основные подходы к построению программной системы контроля и коррекции ошибок в текстах, которая основывается на использовании: статистики искажений; способов и моделей кодирования информации; программных методов контроля информации на основе искусственной и

естественной избыточности; методов, моделей, алгоритмов морфологического анализа.

2. Установлено, что наиболее распространены следующие ошибки: однократные транспозиционные, приводящие к искажению отдельных символов; двукратные смежные транспозиционные; пропуск или добавление дополнительного символа в строке; прочие типы ошибок, т.е. случайные символьные ошибки более высокой кратности. Следовательно, разрабатываемые методы программного контроля и коррекции ошибок должны учитывать установленные характер, тип, природу искажений в обрабатываемой информации.

3. Разработана методика определения объема избыточности, который обеспечивал бы требуемую достоверность информации при применении программных методов контроля и коррекции ошибок, основанных на использовании избыточностей различной природы. Предложены методики синтеза вероятностных процессов, происходящих при контроле и передаче информации; определения количества информации: с учетом происходящих вероятностных процессов; при обычным приеме сообщения; при применении корректирующего кода. В реальных условиях передачи информации для рекомендуемых методов объем избыточности должен быть не менее 0,4.

4. Разработана методика расчета минимального требуемого объема памяти программной системы контроля и коррекции ошибок по справочнику словоформ узбекского языка, которую можно применить и для других естественных языков. Определено, что при примерном количестве словоформ узбекского языка $n = 2^{16}$ целесообразно использовать 16 функций F_i; словоформ русского – $n = 2^{20}$, 20 функций F_i; словоформ английского - $n = 2^{14}$, 14 функций F_i. Для обеспечения вероятности ошибки 2-го рода не более 2^{-20} (10^{-6}) при узбекском языке нужен массив H_i из $2^5 \ldots 2^{20}$ битов.

ГЛАВА II. РАЗРАБОТКА АЛГОРИТМА КОНТРОЛЯ ДОСТОВЕРНОСТИ ЭЛЕКТРОННЫХ ТЕКСТОВ НА ОСНОВЕ МОДЕЛЕЙ УПРАВЛЕНИЯ МОРФОЛОГИЧЕСКИМ СЛОВАРЕМ

2.1. Модели безсловарного морфологического анализа

На современном этапе развития информационных технологий морфологический компонент стал неотъемлемой частью интеллектуальных информационно-поисковых систем (ИПС). В 60-70 гг. XX века все экспериментальные исследования в области машинной морфологии начинались с создания машинного словаря. Не было единого общепринятого формата и структуры такого словаря. Эти обстоятельства имели два последствия: во-первых, все алгоритмы автоматически становились словарнозависимыми, во-вторых, каждый алгоритм разрабатывался под определенный формат словаря [9,10,29-36].

Основная проблема в разработке машинно-ориентированного алгоритма для лингвистических процессоров состоит в объеме исходных данных, используемых программой, то есть в объеме словарей, которые приходится составлять вручную. Исследования в этой области направлены на минимизацию исходных данных. Работы, посвященные морфологии, можно условно разделить на две категории:

- теоретические, в которых представлены описания морфологических законов и формальные модели морфологии естественного языка;

- прикладные, посвященные описанию программно-реализованных систем с морфологическим модулем.

В теоретических работах строятся многоуровневые формальные модели морфологии, в большинстве своем, предназначенные для синтеза. Такие модели морфологического синтеза подразумевают наличие больших словарей со сложной структурой. Они описывают широкий круг морфологических явлений. Многие компоненты этих моделей избыточны для задач машинного анализа (фонетическая реализация слова, акцентная парадигма, большое число словообразовательных аффиксов).

Модели, которые используют словарь, способны дать более полный анализ словоформы (т.е. оперировать большим числом грамматических признаков).

Степень точности такого анализа выше, по сравнению с моделями, которые не используют словаря, однако на пространстве реальных текстов системы, использующие словарь, часто дают сбои. Это обусловлено тем, что не существует полных словарей.

Лексика языка непрерывно пополняется, появляются новые слова. Для каждой предметной области существует своя терминология, свое подмножество лексики языка и включить в общий словарь всю существующую терминологию – невозможно, равно как невозможно перечислить все существующие имена и фамилии, которые имеют регулярное склонение.

Алгоритмы программ, работающих без словаря, используют вероятностно-статистические методы и лексиконы суффиксов или квазисуффиксов, основ или квази-основ, построенных эмпирически. В работе [44] описана работающая модель морфологического анализа, не требующая объемных словарей основ открытых классов слов. Модель разработана в русле инженерной лингвистики. Модель использует следующие лексиконы:

1. Лексикон окончаний и рефлексивов;
2. Лексикон суффиксов;
3. Лексикон квази-корней;
4. Лексикон префиксов;
5. Лексикон баз;
6. Лексикон основ.

Каждой единице такого лексикона приписаны все возможные грамматические характеристики словоформ, частью которой может являться данная единица. Пример единицы лексикона квази-корней:

-ени-
существительное, 11, -е,
существительное, 8, -й,
глагол, -ть;

где 11, 8 - тип склонения.

2.2. Модели морфологического анализа на основе словаря словоформ

Анализ словоформы в безсловарных моделях построен на правилах поиска и сочетания единиц разных лексиконов, что приводит к унификации гипотез. Такой анализ не использует возможности текстов, поступающих на вход системы. В связи с этим предлагается метод, который сводится к эмпирическому сжатию исходного словаря словоформ. Для этого выделяются общие цепочки букв в множестве словоформ, и каждой цепочке букв приписываются все возможные значения грамматических категорий этих словоформ. Эмпирическое сжатие грамматического словаря русского языка приводит к созданию большого числа разрозненных лексиконов разной структуры, каждый из которых требует отдельной процедуры считывания данных. Данный подход к морфологическому анализу нельзя назвать, в полной мере, бессловарным [25-28, 36-39].

Похожий метод основывается на описание вероятностно-статистических методов для создания вспомогательных лексиконов на основе исходного корпуса текстов [42, 43].

Все алгоритмы такого рода имеют одни и те же недостатки:

- не используются точные лингвистические методы анализа;
- большой объем лексиконов;
- вероятностно-статистические методы плохо работают с малой выборкой.

Точность такого анализа намного ниже, чем для систем, работающих со словарем. Эти алгоритмы не позволяют выбирать уникальные грамматические характеристики, хотя в большинстве случаев позволяют построить общую основу или квази-основу для множества словоформ и лемматизировать словоформу.

Наиболее свободная форма анализа разработана в Чикагском Университете [47-50]. Модель позволяет путем статистической обработки большого массива текстов, анализируя частоту встречаемости последовательности символов в словоформах, выделять множество аффиксов и корневых морфем, релевантных для заданного языка. Программа работает с большинством европейских языков,

включая русский. Работа проводилась в рамках научного исследования и не получила прикладного внедрения.

Алгоритмы морфологии построены на самообучении программы на открытых массивах реальных текстов и совмещают два подхода: лингвистический - формализованная грамматика для построения морфологических гипотез и математический - метод корреляции, позволяющий унифицировать морфологическую гипотезу.

Морфологический анализ без словаря является центральной компонентой системы автоматической индексации текстовой базы данных (БД), реализованной в СУБД Oracle. Однако выходным результатом системы является автоматически построенный грамматический словарь основ и связанный с ним индекс документов, предназначенный для полнотекстового поиска по БД. Сущность интеллекта состоит в способности принимать разумные решения в условиях отсутствия полноты данных и фактов. Интеллектуальность системы повышается с уменьшением объема статической информации, используемой в процессе анализа.

В нашем случае, речь идет об использовании лингвистической информации при морфологическом анализе в задачах автоматической индексации текстовых БД. В этой связи выделим основные критерии, отличающие большинство интеллектуальных систем, которых придерживается проектируемый процессор автоиндексации текстов:

- Способность системы объяснить каждой шаг принятых решений. В процессе анализа не используются вероятностные и статистические методы.
- Использование правил и свойств, характеризующих данный предмет анализа. Для построения морфологических гипотез словоформ используется формализованная грамматика и то свойство русского языка, что большая часть грамматических категорий в русском вычисляется из флексии.
- Модульность системы, которая обеспечивает эффективное изменение и пополнение правил и свойств, а также задает возможность настраивать анализатор на другие естественные языки с развитой морфологией.

- Множественность интерпретаций. Анализатор оставляет все омонимы значений словоизменительных категорий.
- Самообучаемость и механизм исправления принятых ранее неверных решений. Объем прочитанных текстов пополняет число словоформ, используемых в процессе анализа, тем самым повышая точность морфологического анализа и позволяя корректировать неправильно построенные основы и значения их грамматических категорий.
- Моделирование интеллектуального поведения человека. В данном случае, речь идет о попытке эмулировать размышления человека, изучающего иностранный язык, перед которым стоит задача классифицировать слова данного языка, в условиях, когда в его распоряжении находится большой массив текстов, некоторые знания о морфологии языка и отсутствует словарь языка, на котором написаны тексты.

2.3. Разработка алгоритма морфологического анализа на основе индексирования

Модель будет рассмотрена на уровне общего описания процессора - взаимодействие его модулей и функциональная схема алгоритма морфологического анализа [29-31].

Схема процесса автоматической индексации представлена на рис.2.1: на вход процесса автоиндексации поступает все множество текстов, хранящихся в базе данных, на выходе формируется словарь основ и таблица соответствий (текст основа), которая отображает поток индексированных текстов.

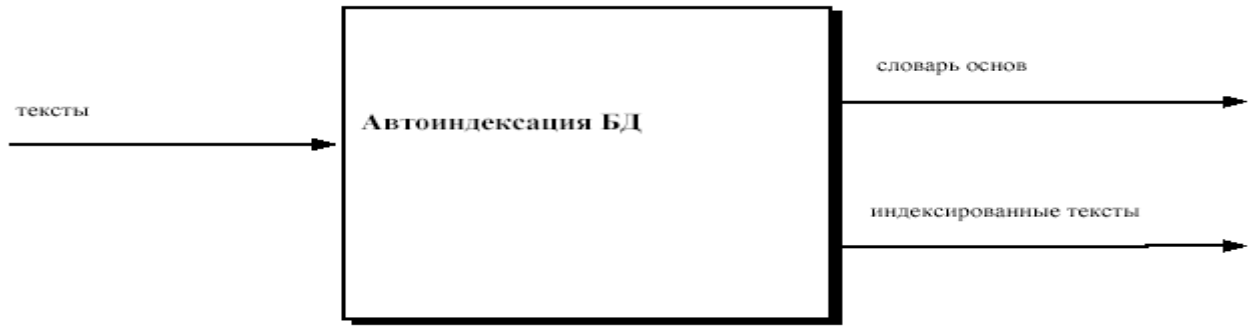

Рис. 2.1 Схема процесса автоматической индексации

Блоки, осуществляющие процесс автоиндексации, представлены на рис. 2.2:

1. Графематический анализ.
2. Морфологический анализ.

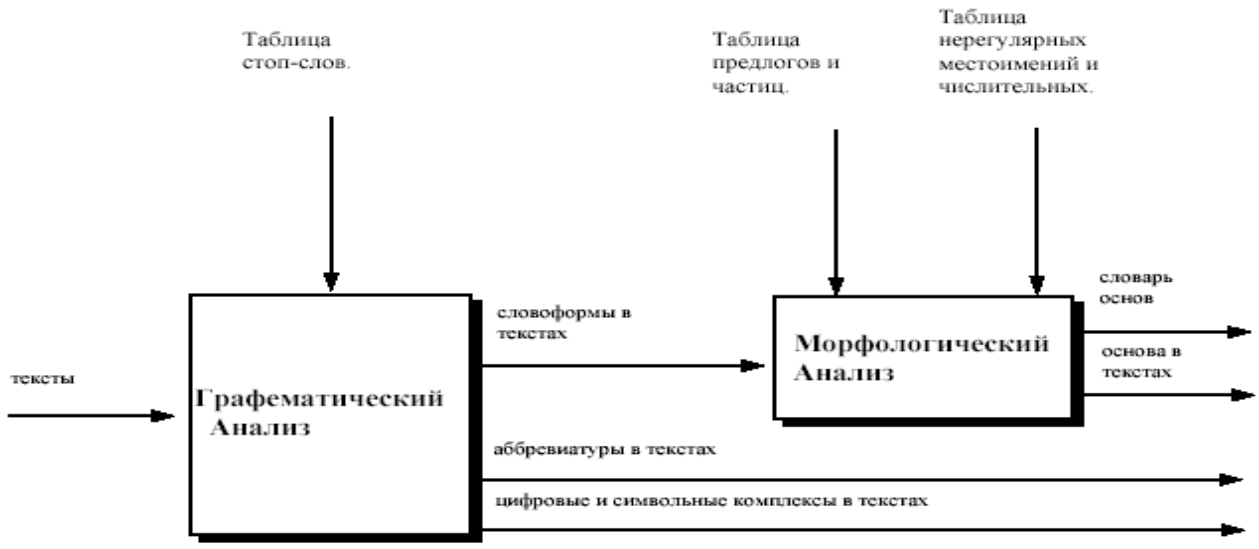

Рис. 2.2. Блоки автоиндексации.

На рис. 2.3 показана схема таблиц для хранения потоков данных, формированных процессами графематического и морфологического анализа. Формируются следующие потоки данных (рис. 2.3):

1. Тексты;
2. Полные словоформы;
3. Аббревиатуры;
4. Цифровые и символьные комплексы;
5. Основы и значения их грамматических категорий;

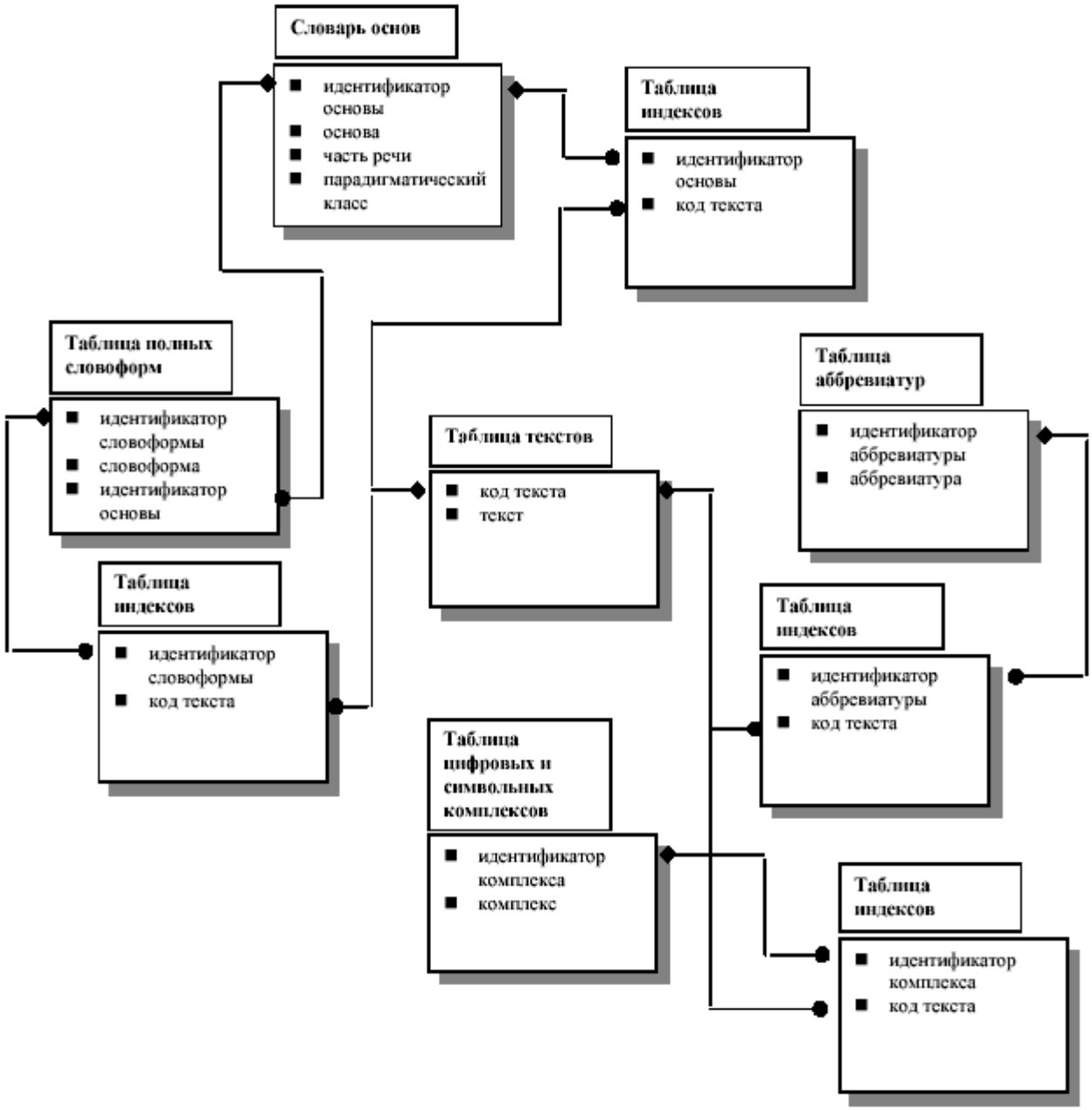

Рис. 2.3. Структурирование потоков данных.

Основная цель блока графематического анализа получить выборку полных словоформ из массива текстов БД. Графематический анализ работает с внешним представлением текста и использует таблицу стоп-слов. В этой таблице хранятся цифры, спецсимволы и частотные слова языка, нерелевантные для поиска по текстам. Графематический анализ выполняет три функции:

1. отсечение стоп-слов в тексте;
2. разбиение данных на три потока;
3. индексация каждого потока.

Единицей графематического анализа является цепочка символов, выделенная с двух сторон пробелами. Выделенная цепочка символов подвергается последовательной обработке эвристическими правилами: отсечь знаки пунктуации, проверить присутствие гласных внутри цепочки, чередование верхнего и нижнего регистров и т.д. В зависимости от результатов обработки полученная цепочка символов направляется в один из трех потоков данных:

- цифровые и символьные комплексы (‘кг’, ‘ст.’, ’12.01.99’);
- аббревиатуры – названия государств, организаций, предприятий (‘РУз’, ‘ЮНЕСКО’, ‘ДорСтройСервис’);
- полные словоформы;

Каждой записи из любого потока ставятся в соответствие коды документов, в которых она встретилась. Первых два потока данных считаются проиндексированными, причем только аббревиатуры являются релевантным поисковым образом.

Графематику можно считать лишь вспомогательным звеном для морфологического анализа. Графематический и морфологический процессы способны проиндексировать массивы текстов независимо от предметной области конкретной базы данных.

Полные словоформы поступают на вход морфологического анализа, цель которого разбить все множество словоформ на подмножества по признаку принадлежности к той или иной лексеме, привести все элементы каждого такого подмножества к уникальной основе, однозначно определить грамматические

характеристики лексемы и проиндексировать тексты по встретившимся в них основам. Блок морфологического анализа использует минимальный объем исходной информации:

- таблицу предлогов;
- таблицу местоимений и числительных, имеющих нерегулярное склонение.

На выходе морфологического анализа формируется словарь основ данной БД, уникальность записи в таком словаре задается тройкой значений [основа, часть речи, парадигматический класс]. Морфологический анализ состоит из трех модулей и соблюдает определенную последовательность действий. Первый модуль содержит статический массив флексий и правила формализованной грамматики русской морфологии, построенной на основе работ [21-23].

Выделяемым парадигматическим классам в модели соответствует восемь типов склонения существительных и прилагательных и шестнадцать типов парадигмы глагола, которым соответствует первое или второе спряжение [28]. Согласно [28] глагольная тема (‘ов’, ‘у’ и т.д.) входит в окончание глагола. В нашем случае вводится термин «расширенная флексия глагола», означающий конкатенацию чередующейся глагольной темы и флексии.

Данный модуль может быть заменен формализованной морфологией любого другого флективного языка. Методы, описанные в модулях два и три, являются универсальными, независящими от языка. Второй модуль, используя правила формализованной грамматики, позволяет строить морфологическое дерево словоформы, в узлах которого хранятся все возможные гипотезы об основах и значениях грамматических категорий словоформы. Морфологические правила делятся на два класса.

Первый класс правил, которые порождают некоторые грамматические характеристики для гипотез, и второй класс правил накладывает определенные ограничения на гипотезы.

Пример правил первого класса: *если гипотеза об основе оканчивается на согласную ряда {‘к’, ‘г’, ‘х’}, то тип склонения равен трем* или *если исходная*

словоформа не оканчивается на гласную, то построить гипотезу о существительном с нуль-флексией.

Пример правил второго класса: *если гипотеза о флексии равна ‘ет’ [3 лицо, ед. ч.] или ‘ю’ [1 лицо, ед. ч.], и гипотеза об основе оканчивается на сегмент первой ступени чередования, то гипотеза о глаголе не верна.*

Традиционно в синтаксических и семантических теориях используется представление языковой структуры с помощью деревьев. В описываемой системе, пожалуй, впервые данный формализм оправданно был применен к морфологии.

Третий модуль содержит метод подбора словоформ на одну лексему, то есть выбор коррелятов для дерева исходной словоформы. После того, как набраны корреляты, для каждой словоформы также строится морфологическое дерево всех возможных гипотез, в результате чего образуется “лес деревьев” [43]. Метод корреляции, принятый в данной работе для применения осуществляет сравнение морфологических деревьев внутри леса и унификацию гипотез. Корреляция проводится по гипотезам основ и значениям классифицирующих грамматических категорий, таких как часть речи, парадигматический класс, спряжение глаголов и род существительных. Значения словоизменительных категорий в корреляции не участвуют. Во время работы корреляции происходит удаление ложных гипотез: ветвей дерева или полного дерева коррелята. Этот модуль позволяет построить уникальную гипотезу об основе и значениях ее грамматических категорий для всех словоформ одной лексемы, найденных в текстах. Метод корреляции очищает лес от ложных коррелятов, оставляя, таким образом, только словоформы, принадлежащие одной лексеме. Уникальная основа, единая для всех словоформ, участвовавших в корреляции, значение части речи и парадигматического класса добавляются в словарь основ.

По сути, основа в словаре репрезентирует лексему. Для унификации гипотезы метод корреляции использует матрицы корреляций. Лесом называется множество деревьев словоформ $F = \{T_1,\ldots,T_j,\ldots,T_n\}$. Множество всех построенных гипотез об основе в F обозначим $U = \{s_1,\ldots,s_i,\ldots,s_m\}$. Параметром корреляции t

называется значение грамматической категории. Матрицей корреляции $A(t) = a_{ij}$ леса F с m гипотезами об основах и n деревьями словоформ называется $m \times n$-матрица, в которой $a_{ij} = 1$, если заданный параметр корреляции t определен для S_i в T_j , и $a_{ij} = 0$ в противном случае.

В процессе корреляции отдается приоритет гипотезам исходной словоформы, на основе которых подбираются корреляты, что позволяет избежать ситуации, когда лес вырождается в пустое множество. Число матриц корреляции внутри одного типа корреляции определяется по числу возможных значений грамматической категории: так, в процессе корреляции по роду существительных для русского будет построено три матрицы, соответствующие трем возможно задействованным в деревьях значениям грамматического рода.

Для каждой матрицы корреляции находится

$$k = \max_{i:a_{i1} \neq 0} \sum_{j=1}^{n} a_{ij}$$

после чего из множества значений k внутри одного типа корреляции также выбирается максимальное значение, которое и соответствует унифицированной гипотезе. Узлы не получившие максимального значения удаляются из деревьев словоформ. Условие $a_n \neq 0$ задает приоритет гипотезам дерева исходной словоформы T_1.

Допустим в прочитанных программой текстах было подобрано два коррелята для исходной словоформы W_1, тогда лес F состоит из трех деревьев словоформ W_1, W_2 и W_3 (рис. 2.4):

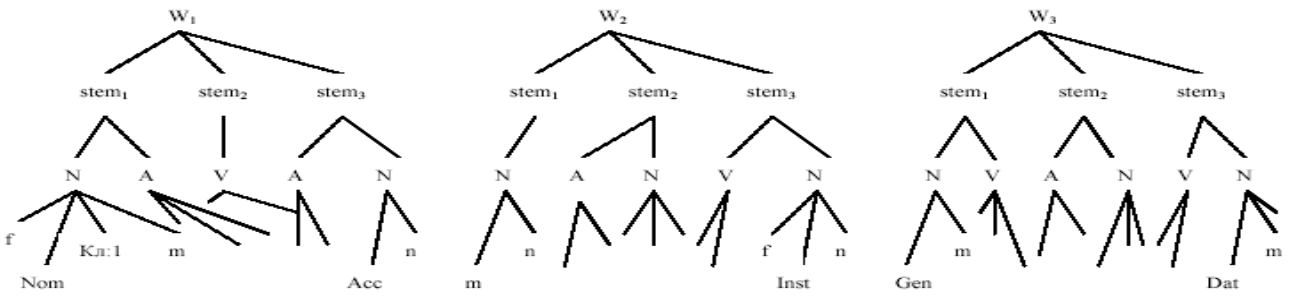

Рис. 2.4. Схема построения деревьев словоформ.

Тогда для корреляции по части речи в соотвествии с методом будут произведены следующие вычисления:

матрица	значение k	максимальное значение внутри типа корреляции
$\text{Noun}\begin{bmatrix}111\\011\\111\end{bmatrix}\bar{1}=\begin{bmatrix}3\\2\\3\end{bmatrix}\Rightarrow\begin{bmatrix}3 & 3\\ stem1 & stem3\end{bmatrix}$		
$\text{Adj}\begin{bmatrix}100\\011\\100\end{bmatrix}\bar{1}=\begin{bmatrix}1\\2\\1\end{bmatrix}\Rightarrow\begin{bmatrix}1 & 1\\ stem1 & stem3\end{bmatrix}$	--- $[3, 3, 1, 1, 1]\Rightarrow\text{Noun}\begin{bmatrix}3 & 3\\ stem1 & stem3\end{bmatrix}$	
$\text{V}\begin{bmatrix}001\\100\\011\end{bmatrix}\bar{1}=\begin{bmatrix}1\\1\\2\end{bmatrix}\Rightarrow\begin{bmatrix}1\\ stem2\end{bmatrix}$		

Результаты такого вычисления позволяют удалить ложные для существительного узлы деревьев словоформ леса *F* (рис.2.5):

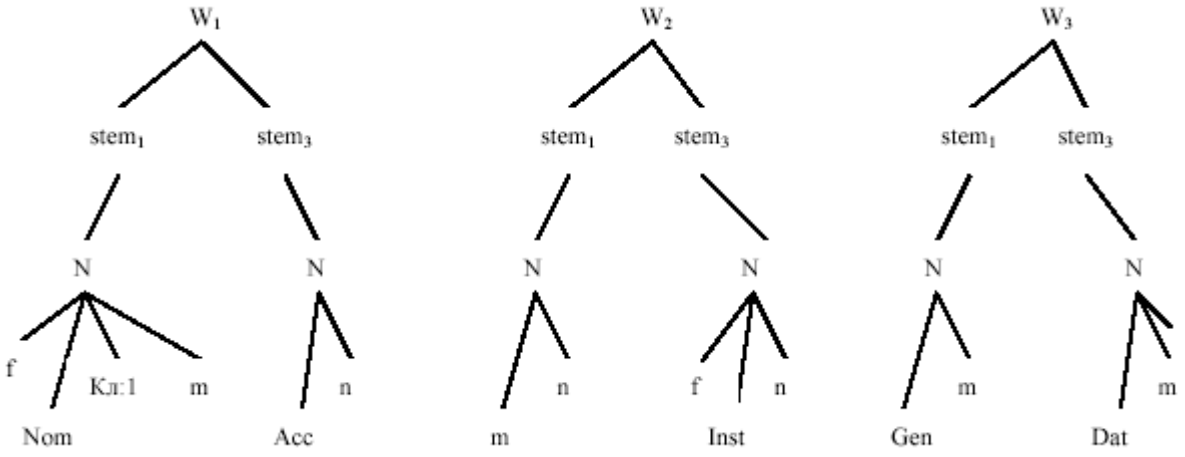

Рис. 2.5. Результат удаления ложных узлов

Теперь проводится корреляция по роду, результат которой также представляется в виде максимального значения внутри типа корреляции. С учетом трех родов существительных (женский, мужской, средний) процесс вычисления максимального значения корреляции представим в следующем виде:

матрица	значение k	максимальное значение внутри типа корреляции
$\mathrm{m}\begin{bmatrix}111\\001\end{bmatrix}\bar{1}=\begin{bmatrix}3\\1\end{bmatrix}\Rightarrow\begin{bmatrix}3\\stem1\end{bmatrix}$		
$\mathrm{n}\begin{bmatrix}010\\110\end{bmatrix}\bar{1}=\begin{bmatrix}1\\2\end{bmatrix}\Rightarrow\begin{bmatrix}2\\stem3\end{bmatrix}$		--- [3, 2, 1] $\Rightarrow \mathrm{m}\begin{bmatrix}3\\stem1\end{bmatrix}$
$\mathrm{f}\begin{bmatrix}100\\010\end{bmatrix}\bar{1}=\begin{bmatrix}1\\1\end{bmatrix}\Rightarrow\begin{bmatrix}1\\stem1\end{bmatrix}$		

После завершения корреляции по роду и удаления не получивших максимального значения узлов гипотеза унифицирована: $W_1[stem_1[N[\hat{E}\ddot{e}:1,\mathrm{m},\mathrm{Nom},\ldots]]]$; $W_2[stem_1[N[\mathrm{m},\ldots]]]$; $W_3[stem_1[N[\mathrm{m},\mathrm{Gen},\ldots]]]$.

Необходимо отметить, что выбор в качестве формализма деревьев, а не кортежей обосновывается возможностью сделать метод корреляции универсальным, независимым от выбранного для анализа естественного языка. Как видно из примеров, ширина дерева произвольна, а высота фиксирована и равна трем для русского языка. Высота дерева, также как и ширина, может изменяться при переходе от одного анализируемого языка к другому и определяется морфологической грамматикой, т.е. существующими зависимостями между грамматическими категориями и их показателями в каждом конкретном языке, что делает использование кортежей затруднительным, а «древесный» формализм сохраняет независимость метода корреляции от морфологических правил рассматриваемого языка.

Рассмотрим реальный пример. Визуальный интерфейс программы морфологического анализа позволяет наблюдать состояние леса до корреляции и после, как это показано на примере анализа глагола ‘текут’ (Рис.2.6 и Рис.2.7).

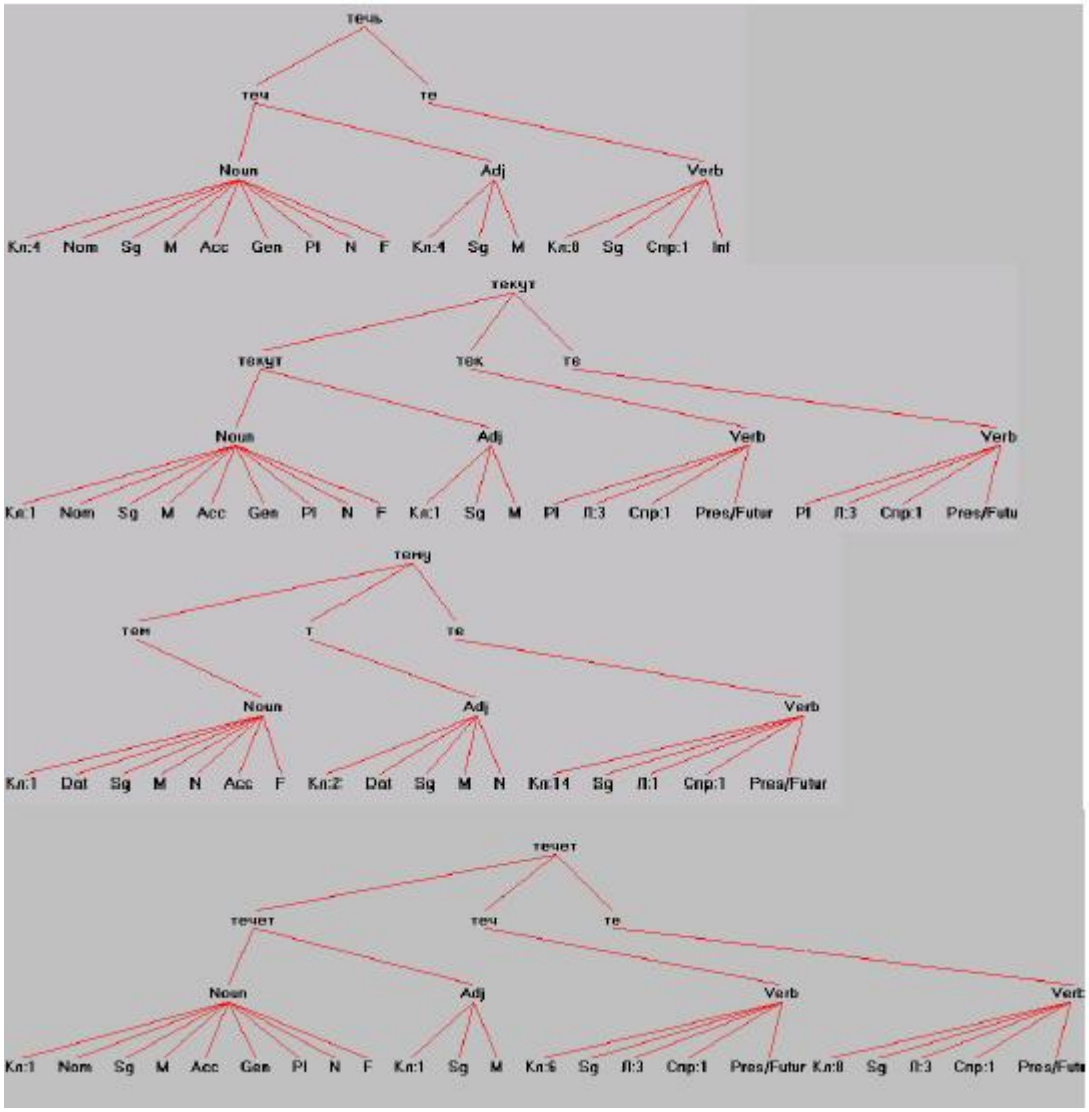

Рис. 2.6. Подбор коррелятов и построение леса деревьев возможных гипотез для глагола ‘текут’:

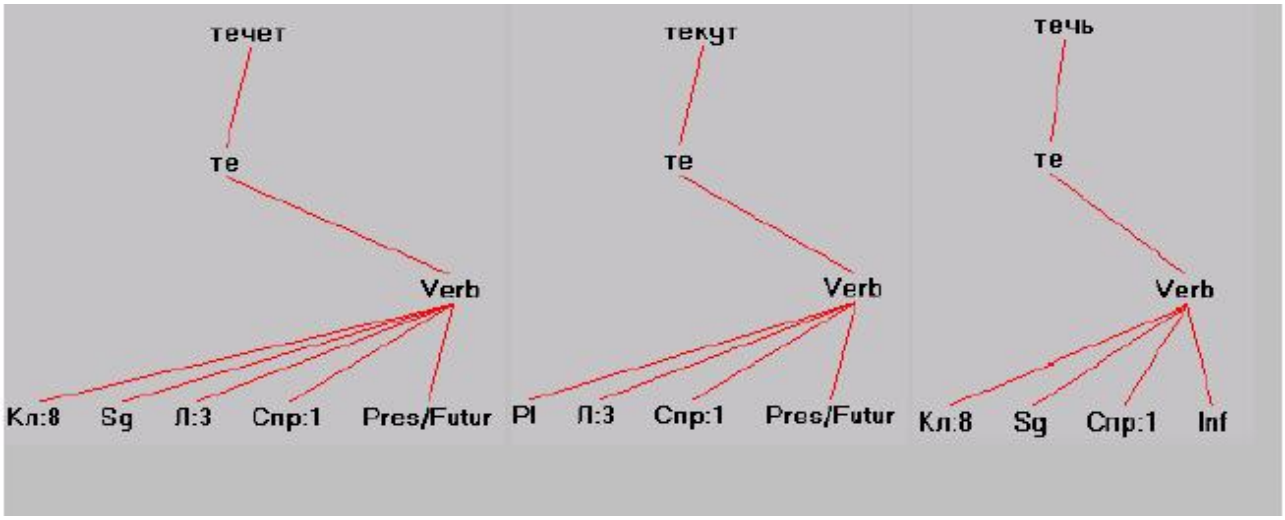

Рис. 2.7. Применение метода корреляции (унификация гипотезы и удаление ложных коррелятов):

В результате остается три дерева коррелятов с уникальными гипотезами об основе, части речи и грамматических характеристиках.

Последовательность шагов (Д1..Д13) алгоритма морфологического анализа без словаря представлена на рис.2.8.

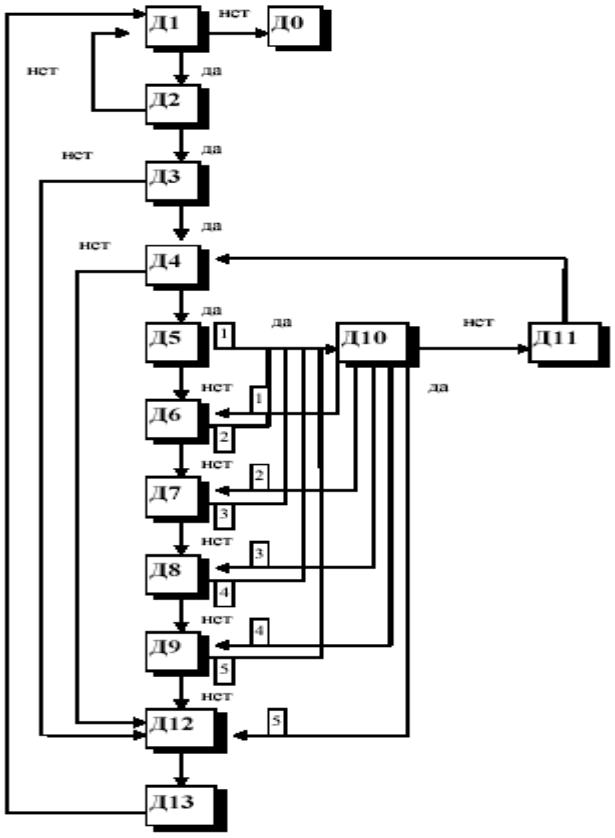

Рис.2.8.

Д0. Выход из программы.

Д1. Выбрать из таблицы полных словоформ непроиндексированную словоформу, то есть словоформу, для которой еще не построена основа (ДА: словоформа выбрана; НЕТ: все словоформы в таблице проиндексированы).

Д2. Проверить, что данная словоформа не является предлогом или местоимением. Построить дерево всех возможных гипотез для данной словоформы. (ДА: не является; НЕТ: является)

Д3. Выбрать из таблицы полных словоформ (рис.8) словоформы на одну лексему. Создать список коррелятов. (ДА: корреляты выбраны; НЕТ: список коррелятов пуст)

Д4. Если список коррелятов непустой, то построить деревья всех возможных гипотез для каждого коррелята.

Д5. Провести корреляцию по гипотезам основ.

Д6. Провести корреляцию по значениям части речи.

Д7. Провести корреляцию по значениям спряжения глагола.

Д8. Провести корреляцию по значениям рода существительных.

Д9. Провести корреляцию по значениям парадигматического класса (для Д5 - Д9 ДА: корреляция прошла успешно, то есть в деревьях словоформ были

обнаружены ложные ветви и удалены; НЕТ: корреляция прошла неуспешно, то есть ложных ветвей нет).

Д10. Проверить, что корреляция не привела к удалению полного дерева (дерева коррелята) из леса. (ДА: не привела; НЕТ: привела)

Д11. Удалить ложный коррелят из списка коррелятов.

Д12. Выбрать уникальную основу и ряд грамматических характеристик к данной основе. Проиндексировать тексты, то есть выбрать для построившейся тройки [основа, часть речи, парадигматический класс] коды текстов, в которых встретились словоформы, принадлежащие данной основе.

Д13. Применить метод распределения элементов пересеченных множеств коррелятов.

Несмотря на появление объемных лексиконов для многих европейских языков и все возрастающую популярность словарного анализа, системы морфологического анализа без словаря не теряют своего прикладного значения.

В задачах автоматической индексации изложенные выше алгоритмы позволяют формировать грамматические словари, являющиеся точным отображением лексики проиндексированных документов. Бессловарная морфология сохраняет свою актуальность в задачах автоматического пополнения лексиконов. Точность такого анализа выше, чем стандартная процедура предсказания по конечной последовательности символов в слове Использование деревьев для представления морфологической структуры словоформы и унификация гипотезы роднит задачи морфологического и синтаксического анализа, демонстрируя общность формализма и алгоритмических методов на разных уровнях лингвистического анализа. Тестирование программы, разработанной на основе полученной методики, показало работоспособность предложенной системы автоматической индексации. Метод корреляции, разработанный для настоящей задачи, позволяет выбирать уникальные гипотезы и строить словарь основ при сравнительно небольшой выборке.

2.4. Проектирование алгоритма морфологического лексикона

При проектировании алгоритма морфологического лексикона рассматриваются явления, которые могут быть как внутренними, относящимися к терминальным единицам и связям между ними, так и внешними, т.е. универсальными структурными законами. Существующий набор явлений в современной теоретической лингвистике намного шире, однако не все из них хорошо формализуемы и могут быть использованы в прикладных моделях.

Все языковые средства, которые использует разрабатываемая система для определения синтаксических понятий, являются либо свойствами самого объекта, т.е. предложения естественного языка, либо свойствами его элементов, т.е. словоформ и знаков пунктуации (операторов). Синтаксические понятия, по существу, представляют собой функции, где параметрами служат языковые средства, а сами функции используются в условиях грамматических стратегий или правил. Ниже приведены пять языковых средств синтаксического анализа, используемых при разработке алгритма морфологического лексикона:

1. *Словоизменительные морфологические средства*. Для языков с развитой морфологией, каким является узбекский - это основной способ материализации синтаксических связей. Словоформа $w1$ морфологически зависит от словоформы $w2$ по морфологической категории C, если граммема (значение грамматической категории) g категории C, характеризующей $w1$, выбирается в зависимости от некоторого свойства f словоформы $w2$. Словоформа $w2$ называется контролером морфологической зависимости, а $w1$ - ее мишенью [40]. Другими словами, один элемент предложения подстраивается под другой, т.е. принимает грамматическую форму, продиктованную вторым элементом. Показателем морфологической зависимости в узбекском служит флексия, т.к. граммемы в узбекском обычно приписаны флексии, что позволяет в некоторых случаях обнаружить зависимость между двумя словоформами, отсутствующими в словаре, (например, "мен-инг китоб-им"). Если категория C, по которой наблюдается морфологическая зависимость, выражается в вершине, налицо вершинное маркирование, если же эта категория выражается в зависимой

словоформе - зависимостное маркирование [40]. В узбекском языке граммемы многих форм омонимичны ('гуллар' = [с.,о.п.,мн.], [г.,б-в.в.,пол.ф.]), что создает определенные трудности в процессе анализа. Неоднозначность граммем в ходе автоматического синтаксического анализа иногда приводит к возникновению синтаксической омонимии и построению альтернативного синтаксического варианта. Падежная омонимия с номинативом часто приводит к неоднозначному определению правой границы сегмента и, как следствие, к построению альтернативной структуры сегментации (графа сегментов). Парадокс или скорее взаимовлияние двух уровней анализа морфологического и синтаксического состоит в том, что граммема, являясь эффективным средством поиска морфологической зависимости, которая служит одним из способов реализации синтаксического отношения, может быть однозначно проинтерпретирована только вследствие фиксации этого отношения.

2. *Селективные признаки.* Классифицирующие (селективные) признаки приписываются лексемам в грамматическом словаре, в отличие от граммем, которые вычисляются, исходя из парадигматического класса, для каждой словоформы на этапе морфологического анализа. Наиболее важной для синтаксиса является классификация лексем по категориальным (частеречным) признакам: существительное, глагол, прилагательное, и т.д.

3. *Служебные слова.* Послелоги, союзы и союзные слова, вспомогательные компоненты аналитических форм, частицы и т.д. Средства, которые служат в качестве опорных точек анализа. Так, союз может быть использован для определения поверхностного типа сегмента, или вспомогательный компонент аналитической формы содержит недостающие предикату граммемы, или предлог оформляет актант глагола.

4. *Знаки препинания* (операторы). Запятая, тире, точка, вопросительный знак, и т.д. Это средство не выделяется в теоретических описаниях, так как теоретический синтаксис имеет дело больше с устным языком, чем с письменным, к тому же не все письменные языки, в отличие от узбекского, имеют жесткие правила расстановки знаков препинания. В первую очередь, операторы

определяют границы, как сегментов, так и всего предложения. В теоретических работах принято выделять интонацию как средство синтаксического анализа. Действительно, операторы в письменном тексте являются частичным выражением подмножества синтаксических случаев, характеризующихся интонацией в устном языке [24]. Такие случаи применения интонации для различения синтаксических связей не фиксируются операторами в письменной форме, поэтому идеальный синтаксический процессор должен решить эту проблему через понятие синтаксической омонимии, построив две равноправных синтаксических структуры предложения.

5. *Порядок слов.* Линейное расположение слов в предложении играет особую роль в изолирующих языках (китайский) и является основным средством для выражения синтаксических отношений в этих языках. Наряду с селективными признаками порядок слов имеет доминирующее значение в проектировании синтаксических анализаторов языков с бедной морфологией (английский). Во многих системах английского синтаксиса порядок слов задает направление поиска хозяина или слуги для каждого класса лексем и типа связи. Для узбекского языка это средство анализа также имеет большое значение, так как в узбексом языке обычен жесткий порядок слов. Например, зависимый член предшествует главному, определение стоит за определяемым словом ('стол-нинг оёг-и', 'ота-нинг ўғл-и'); существительное предшествует послелогу ('стол олдида'); в 90% случаев определение, выраженное прилагательным или местоименным прилагательным, стоит до существительного ('катта чиройли стол', 'кекса одам'). Порой статистическое расположение синтаксических вершин и их зависимых позволяет разделить все типы синтаксических отношений на три типа: левоветвящиеся (прилагательное существительное: 90%), правоветвящиеся (генитивное определение: 100%) и смешанные (слабые актанты глагола: 50%/50%). Подобные эмпирические распределения могут эффективно использоваться в прикладных моделях. В лингвистической типологии эмпирически установлена универсальная классификация языков мира: языки левого (японский) и правого ветвления (русский и английский и узбекский).

Правда, эта классификация, в основном, строится на статистическом распределении фразовых категорий в линейном порядке предложения, к которым относятся именные (NP), предложные группы (PP) и клаузы (некоторые виды сегментов: придаточные определительные, причастные обороты, и т.д.). Другая синтаксическая классификация оперирует линейным порядком основных членов предложения: подлежащее (subject), сказуемое (verb) и дополнение (object). Английский относится к языкам Subject Verb Object (SVO) порядка. Узбекский относится к языкам SOV порядка. В узбекском предложении ‘Фермер хўрозни тутди’ любое изменение порядка слов не ведет к изменению смысла всего высказывания, хотя и нарушается грамматическая правильность ('Хўрозни фермер тутди'), но в русском переводном эквиваленте (‘Фермер поймал утенка’) возможено 3! перестановок, сохраняющих как общий смысл высказывания, так и грамматическую правильность, т.е. в русском варианте данного предложения возможны любые комбинаторные порядки: SVO, SOV, OVS, и т.д. Таким образом, линейный порядок предложения в автоматическом синтаксическом анализе используется как указатель наиболее вероятного направления поиска слуги или хозяина, и только в редких случаях как обязательный критерий установления синтаксической зависимости.

Ниже опишем процедуры, оперирующие изложенными языковыми средствами, только в рамках их приложения в синтаксическом анализаторе:

1. Согласованием является результат выполнения пересечения векторов граммем двух словоформ, где ожидаемый результат пересечения определяется категориальными признаками словоформ. Согласование может быть полным или частичным.

Полное согласование:

а) $VA \cap VN = [c, Sg, g] \parallel [c, Pl]$, где VA - вектор граммем полного прилагательного, причастия или местоименного прилагательного; VN - вектор граммем существительного; $c \in C$ = [им., рд., вн., дт., тв., пр.] - значение падежа; Sg (ед. ч.) и Pl (мн. ч.) - значения грамматического числа; $g \in G$ = [мр., жр., ср.] - значение грамматического рода.

b) VSnom∩VP = [p≠∅, n] || [g] || [p=∅, Pl], где VSnom - вектор граммем подлежащего, выраженного существительным или местоимением в именительном падеже; VP - вектор граммем сказуемого, выраженного финитной формой глагола или краткой формой прилагательного или причастия; p ∈ P = [∅, 1л., 2л., 3л.] - значение грамматического лица; n ∈ N = [Sg, Pl].

Частичное согласование:

a) VA∩VN = [c], такой тип согласования используется в дуальных конструкциях (например, "қизил стол ва стул" или "кук ва қизил коптоклар"), в тех случаях когда еще не построены сочинительные группы. Применение частичного согласования в этих конструкциях зависит полностью от грамматического описания, принятого в прикладной модели. Альтернативный вариант анализа дуальных конструкций состоит в предварительном поиске сочинительных групп, вычисления граммем группы и сведения проверки согласования при последующем установлении атрибутивной связи (именной группы) к полному согласованию типа (a).

b) VA1∩VA2 = [c], VN1∩VN2 = [c], VP1∩VP2 = [p≠∅, n] || [Imptv, n] || [Inf] || [g] || [p=∅, Pl], где Imptv - императив, Inf - инфинитив.

Подобного рода согласование используется для определения сочинительных конструкций в русском языке.

2. Примитивное управление с помощью вектора M, определенного в словаре для каждой лексемы L, способной управлять словоформой X. Вектор M лексемы L содержит значения селективных признаков и/или граммемы словоформы X. Вектор M ⊂ M| = [предлог, подчинительный союз, инфинитив, им., рд., вн., дт., тв., пр.]. Управлением называется пресечение вектора M лексемы L с вектором граммем словоформы X или с значением селективных признаков словоформы X. Явление примыкания и конгруэнтности, а также более сложные случаи управления, не используются в предлагаемых моделях синтаксических анализаторов и считаются прерогативой этапа первичного семантического анализа [39].

3. Объединение значений селективных признаков в более крупные единицы, использующиеся в синтаксических моделях. Предикат в предложении может быть выражен словоформой с значением части речи ps ∈ PS = [финитная ф. гл., кр. прил., кр. прич., предикатив]. При построении атрибутивной связи AN A может быть выражено словоформой с значением части речи a ∈ A = [полное прилагательное, полное причастие, местоименное прилагательное], а N может быть выражено словоформой с значением части речи n ∈ N = [существительное, местоимение, субстантивированное прилагательное].

В лексиконах изложенные выше процедуры обычно оформляются в виде программных функций, которые служат для проверки и установления возможного синтаксического отношения. Далее изложенные процедуры объединяются в более крупных модулях анализа, которые отражают грамматические правила и стратегии. Каждое грамматическое правило устанавливает один тип синтаксического отношения R(A, B) между двумя единицами анализа и однозначно задает вершину. Число используемых типов отношений, а также их названия, зависит от прикладной модели и конкретной системы.

Т.к. для узбекского языка эта область пока малоизученна, в данной работе предпринята попытка адаптировать синтаксические отношения русского языка к узбекскому. В роли единиц анализа, на месте A и B, где A - вершина, а B - зависимое, могут выступать как отдельные словоформы, так и целые группы (фразовые составляющие); заполнение A и B во многом зависит от синтаксического аппарата, принятого в анализаторе для описания структуры. Использование грамматических правил задает прозрачность архитектуры процессора и обеспечивает устойчивость системы к изменениям.

Выводы к главе 2.

1. Проведен анализ исследований в области машинной морфологии. Опредена основная проблема в разработке машинно-ориентированного алгоритма для лингвистических процессоров, состоящая в объеме исходных данных, используемых программой, то есть в объеме словарей, которые приходится

составлять вручную. Исследования в этой области направлены на минимизацию исходных данных.

2. Построены модели морфологического синтеза, подразумевающие наличие больших словарей со сложной структурой. Выявлены избыточные элементы моделей, описывающих широкий круг морфологических явлений для задач машинного анализа - фонетическая реализация слова, акцентная парадигма, большое число словообразовательных аффиксов. Описана работающая модель морфологического анализа, не требующая объемных словарей основ открытых классов слов. Модель разработана в русле инженерной лингвистики.

3. Рассотрены вопросы построения алгоритмов морфологии по принципам самообучения программы на открытых массивах реальных текстов при совмещении двух подходов: лингвистического – формализованной грамматика для построения морфологических гипотез и математического - метода корреляции, позволяющего унифицировать морфологическую гипотезу.

4. Предложена методика использования лингвистической информации при морфологическом анализе в задачах автоматической индексации текстовых БД. Выделены основные критерии, которых придерживается проектируемый процессор автоиндексации текстов.

5. Представлены синтаксические понятия, представляющие собой функции, где параметрами служат языковые средства, а сами функции используются в условиях грамматических стратегий или правил. Приведены языковые средства синтаксического анализа, используемых при разработке алгоритма морфологического лексикона. Разработаны процедуры, оперирующие изложенными языковыми средствами только в рамках их приложения в синтаксическом анализаторе.

6. Адаптированы синтаксические отношения русского языка к узбекскому. В роли единиц анализа выступают как отдельные словоформы, так и целые группы (фразовые составляющие). Использованы грамматические правила, задающие прозрачность архитектуры процессора и обеспечивающие устойчивость системы к изменениям.

ГЛАВА III. РЕАЛИЗАЦИЯ АЛГОРИТМОВ КОНТРОЛЯ ДОСТОВЕРНОСТИ ТЕКСТОВОЙ ИНФОРМАЦИИ НА ОСНОВЕ УПРАВЛЕНИЯ МОРФОЛОГИЧЕСКИМ СЛОВАРЕМ

3.1. Реализация алгоритмов контроля электронных текстов на узбекском языке

Так как узбекский язык относится к агглютивным, морфологический анализ без словаря будет чрезвычайно сложным и неточным. В настоящий момент ведутся работы по созданию системы выделения лемм из электронных текстов на основе статистических данных. Наличие словаря лемм упрощает морфологический анализ слов, повышает точность. Каждой лемме в словаре возможно добавление перевода, что позволит построить машинный переводчик. При статическом анализе или анализе без словаря для построения переводчика всё равно необходимо создание словаря.

На рис. 3.1. приведена схема анализа текстов на узбекском языке для построения словаря словоформ, управление которым позволит упростить предсказание основы слова и его части речи на основе анализа аффиксов ввиду агглютинативности языка. В анализаторах русского языка, к примеру, предсказание основано на нахождении слов с одинаковым окончанием, что является очень не точным, тем более такой метод не применим для узбекского языка из-за наличия словообразовательных префиксов, например давлат – богатство – сущ., ба-давлат – богатый – прил.

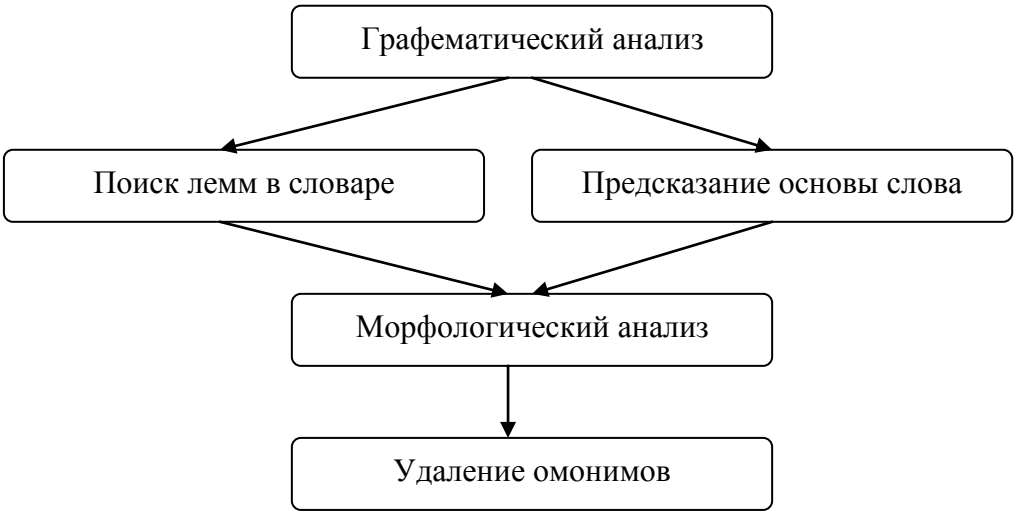

Рис. 3.1. Общая схема действий анализа

Согласно схеме графематический модуль разбивает исходный текст документа на предложения и слова, выстраивая внешнее представление текста

Модуль предсказания позволяет определить часть речи для слов не найденных в словаре. Базовая морфологическая функция получает на вход слово и выдает его лемму (словарную форму: "столом" -> "стол") и морфологическую информацию (число, падеж, ...). Блок удаления омонимов удаляет неправильно найденные леммы в словаре, а также одинаковые слова которые были найдены в словаре, а также предсказаны.

Для ускорения поиска слов в словаре была предпринята попытка сначала предсказывать основу. К сожалению этот способ пропускает множество омонимов, которые могут быть найдены в словаре при обычном поиске, поэтому пришлось от него отказаться.

Графематический анализ. Графематический анализ (далее графематика) - достаточно простая программа, выполняющая первые предварительные действия над текстом. На вход графематике подается текст в кодировке Unicode, на выходе строится графематическая таблица, в которой на каждой строке стоит слово или разделитель из входного текста.

Графематическая таблица состоит из двух столбцов:

Кусок входного текста	**Графематические дескрипторы**
пахта	
очилди	
,	Z
терим	
бошланади	
.	G

Возможны следующие графематические дескрипторы:

Назв.	**Объяснение**	**Пример**
Z	Присваивается запятой	,
N	разделитель, но не знак препинания	';:
G	конец предложения.	

Выделение лемм. Морфологический компонент осуществляет морфоанализ и лемматизацию узбекских словоформ (лемматизация – приведение текстовых форм слова к словарным; морфоанализ – приписывание словоформам морфологической информации).

В системе используются два типа словаря:

- обычный словарь узбекских слов;
- словарь предсказанных слов.

В дальнейшем планируется дополнить словари следующими типами:

- Словарь имен собственных (например: Алишер, Ибрагим, Иванов);
- Словарь географических слов (например: Ташкент, Узбекистан).

При лемматизации для каждого слова входного текста морфологический процессор выдает множество морфологических интерпретаций следующего вида:

- лемма (всегда пишется маленькими буквами);
- морфологическая часть речи;
- множество наборов граммем.

Лемма – это нормальная форма слова. Например, для существительных – это единственное число, именительный падеж.

Морфологическая часть речи определяется значением, указанном в словаре, за исключением случая предсказания. Граммема – это элементарный морфологический описатель, относящий словоформу к какому-то морфологическому классу, например, словоформе “гуллар” будут приписаны следующие значения:

1. гуллар – гулламоқ - цвести (гл., будущее возможное время)
2. гуллар – гул - цветок (сущ., мн.ч.)

Таким образом, морфологический анализ выдает два варианта анализа словоформы “гуллар” Поиск лемм в словаре производится последовательным усечением словоформы на один символ. В словаре также ищутся словосочетания максимум из трёх слов. В словаре ищутся всё возможные совпадения с урезанной словоформой. На данном этапе также осуществляется проверка слова на присутствие словообразовательных префиксов.

Список словообразовательных префиксов, учитываемых системой следующий:

Префикс	Образуемая часть речи	Часть речи, от которой образуется
Ҳам-	Сущ	-
Сер-	Прил	Сущ
Ба-	Прил	Сущ
Бе-	Прил	Сущ
Бад-	Прил	Сущ
Хуш-	Прил	Сущ
Бар-	Прил	Сущ
Но-	Прил	Прил

Это позволяет уже на этапе поиска лемм находить слова, отсутствующие в словаре. После поиска в основном словаре происходит предсказание основы и поиск в словаре предсказанных слов. Все найденные слова получают определённый “вес”. Слова найденные в словаре – наивысший, слова найденные с помощью выделения префиксов – ниже. Самый низкий “вес” имеют предсказанные леммы. Все полученные данные, включая результаты работы графематического анализа заносятся в таблицу следующего вида:

Кусок входного текста	Лемма	Перевод	Дескриптор	Омоним
Ўқитувчи	ўқитувчи	учитель	С	1
Ўқитувчи	ўқитмоқ	учить	Г	1
Ўқитувчи	ўқимоқ	учить	Г	1
Ўқитувчи	ўқ	пуля	С	1
синфга	синг	класс	С	2
синфга	синмоқ	ломаться	Г	2
кирди	кирмоқ	входить	Г	3
кирди	кир	грязь	С	3
кирди	кир	грязный	П	3
.	.	.	G	4

Полученные данные обрабатываются блоком морфологического анализа.

Предсказание и самообучение

Предсказание основы слова и части речи основывается на том факте, что в узбекском языке существует строгая аффиксация. При этом для определённых частей речи используются только определённые аффиксы или их комбинации. В настоящий момент модуль позволяет предсказывать существительные и глаголы.

Предсказание существительных

Примерная модель аффиксации существительного в узбекском языке следующая:

Лемма + [S] + [M] + [L] + [P] , где

S - словообразовательные аффиксы

M - аффикс множественности [лар]

L - аффикс личной принадлежности [и(м), и(нг), с(и), и(миз), и(нгиз)]

P - падежные аффиксы [нинг, ни, га (ка, қа), да, дан]

Предсказание основы основано на нахождении минимум двух аффиксов из категории M, L или P. При этом найденные аффиксы должны находиться именно в данной последовательности. Возможные варианты: M-L, M-P, L-P.

Примеры:

- китобларим → китоб + лар + им = лемма + M + L
- уйимга → уй + им + га = лемма + L + P

Предсказание глаголов

Модель аффиксации глагола следующая:

Лемма + [S] + [Z] + [O] + [N] + [V] + [M] + [D], где

S - словообразовательные аффиксы

Z - залоговый показатель [и(л), и(н), и(ш)]

O - показатель отрицания [ма,]

N - показатель наклонения или модуса [й, йин, ай, айин и т.д.]

V - временной показатель [ган, яп,ар и т.д.]

M - показатель лица вместе с числом [м, манн, нг, сан, ди ,миз, нгиз и т.д.]

D - аффикс вопросительности [ми]

Предсказание основы основано на нахождении двух аффиксов из категории V и M. При этом найденные аффиксы должны находиться именно в данной последовательности.

Пример: ёзаман → ёз + а + ман = лемма + V + M

Следует отменить, что все аффиксы должны быть идентифицированы. Проведённые исследования показывают высокую точность данного метода предсказания.

Все предсказанные слова сохраняются в словаре, что даёт в дальнейшем находить эти же слова, но с одним аффиксом или вообще без них. Происходит как бы “обучение анализатора”

Морфологический анализ

Морфологический анализ реализован для следующих частей речи:

- Существительные
- Глаголы
- Местоимения
- Прилагательные
- Числительные
- Причастия

На первом этапе производится анализ словообразовательных аффиксов. В зависимости от найденных аффиксов порождаются новые морфологические интерпретации с изменением леммы. У новых омонимов снижается “вес”

Например:

Олмазор:

олма – сущ. – существующая лемма

олма + зор - олмазор – сущ. Новая лемма - олмазор

Китобча:

китоб – сущ – существующая лемма

китоб + ча – сущ. Новая лемма - китобча

китоб + ча – нареч. Новая лемма - китобча

Анализ различных частей речи происходит отдельно. При анализе также возможно порождение новых омонимов. Например если в глаголе будет найден аффикс “-ган”, который одновременно выполняет функцию прошедшего времени для глаголов, а также словообразовательную для причастий прошедшего времени.

Идентификация аффиксов начинается с первого символа после окончания леммы словоформы. Для глаголов и прилагательных также проверяются предыдущие и последующие слова. Например, два слова “жуда” – наречие и

“чиройли” – прилагательное, будут объединены в одно “жуда чиройли” – “очень красивый” с пометкой в граммеме – превосходная степень прилагательного.

Снятие омонимии

Удаление неправильных морфологических интерпретаций происходит в два этапа:

1. Проверка всех слов на правильность аффиксов.
2. Удаление омонимичных слов с наименьшим весом

Проверка на правильность аффиксов осуществляется специальной процедурой, которая просто перебирает возможные варианты аффиксов у определённых частей речи. Если проверка не пройдена, у словоформы удаляется пометка о части речи и снижается “вес” до самого наименьшего.

На втором этапе сверяется “вес” у всех морфологических интерпретаций одного слова. Омонимы с нулевым “весом” удаляются.

3.2. Выбор программной среды для реализации системы контроля достоверности электронных текстов

Visual Basic .NET— это один из самых эффективных инструментов для ускоренного создания приложений для Microsoft Windows и интернета. Visual Basic .NET идеально подходит как для разработчиков, уже работающих на языке Visual Basic, так и для тех, кто хочет создавать приложения с использованием платформы Microsoft .NET. В составе Visual Basic .NET поставляется мощная интегрированная среда разработки с усовершенствованными визуальными конструкторами, которая позволяет создавать приложения за короткое время. Производительность создаваемых приложений значительно увеличена.

Язык Visual Basic унаследовал дух, стиль и отчасти синтаксис своего предка — языка Бэйсик, у которого есть немало диалектов. В то же время Visual Basic сочетает в себе процедуры и элементы объектно-ориентированных и компонентно-ориентированных языков программирования.

Среда разработки VB включает инструменты для визуального конструирования пользовательского интерфейса. Разработчики могут применять Visual Basic .NET 2005 для достижения следующих целей.

- Решение актуальных задач, связанных с разработкой приложений для Windows и интернета. Простые и понятные визуальные конструкторы форм Windows Forms и Web Forms обеспечивают одинаковые способы разработки интерфейсов как для полнофункциональных настольных приложений, так и универсальных веб-приложений. Средства развертывания приложений позволяют избежать проблемы совместимости DLL-библиотек (DLL Hell). В составе библиотек доступа к данным ADO.NET имеются развитые классы и компоненты, на основе которых можно реализовать самые разные сценарии работы с данными.
- Создание перспективных приложений. Благодаря возможности разработки приложений для мобильных устройств разработчики могут использовать свои навыки в целях создания мобильных веб-приложений. С помощью шаблона проекта веб-службы XML можно создавать удаленные компоненты бизнес-

логики, причем так же просто, как и обычный класс Visual Basic. Использование дополнительных объектных структур позволяет сэкономить время за счет повторного использования кода и интерфейсов Windows Forms.

- Обновление и повторное использование имеющегося кода Visual Basic. Усовершенствованный мастер обновлений Visual Basic .NET служит для преобразования существующих приложений, выполненных в Visual Basic 6.0, в проекты Visual Basic .NET и могут взаимодействовать с компонентными приложениями Visual Basic предыдущих версий.

Обычно Basic ассоциируется с чем-то очень простым в освоении и использовании для программирования. Это действительно так. На заре компьютерных технологий язык BASIC был создан для простых программ и использовался в качестве учебного языка для первых шагов при изучении основ программирования с последующим переходом на более сложные и универсальные языки. С прогрессом компьютерных технологий развивался и BASIC. В настоящее время версия Visual Basic даёт возможность решать любые современные задачи разработки приложений. При этом этот язык остаётся достаточно простым в освоении, став в то же время мощным современным языком программирования.

Исполнительная среда (runtime) всегда присутствовала в Visual Basic, поэтому следующее утверждение поначалу выглядит несколько странно. Итак, одним из самых серьезных новшеств VB.NET является наличие исполнительной среды CLR (Common Language Runtime), общей для всех языков .NET. Хотя на первый взгляд CLR напоминает обычную библиотеку времени выполнения наподобие библиотеки C MSVCRTXX.DLL, библиотека VB MSVBVMXX.DLL имеет значительно большие размеры и обладает гораздо большими возможностями. По этой причине написание программ, в полной мере использующих CLR, больше походит на программирование для API новой операционной системы. Поскольку все языки .NET используют одну и ту же среду CLR, необходимость в исполнительных средах для отдельных языков отпадает. Более того, код, предназначенный для выполнения в CLR, может быть

написан на любом языке и с одинаковым успехом использоваться во всех языках, соответствующих спецификации CLR. В этом проявляется главное отличие .NET от Java: на платформе .NET можно использовать любой язык при условии, что он соответствует спецификации CLR. Программа, написанная на Java, работает на любой платформе (по крайней мере теоретически — на практике возникают проблемы), но при условии, что она написана именно на Java. Вероятно, именно языковая интеграция станет одной из составляющих успеха .NET. В частности, код VB может использоваться в программах, написанных на C#, и наоборот, причем это не потребует дополнительных усилий со стороны программиста.

У программистов, работающих на Visual Basic, всегда возникали проблемы с утечкой памяти из-за так называемых циклических ссылок (ситуация, при которой объект A ссылается на объект B, а объект B ссылается на объект A). Если появление циклических ссылок было обусловлено логикой программы, компилятор VB не распознавал их, в результате чего память, занимаемая этими объектами, не освобождалась. Система сборки мусора, встроенная в .NET CLR, решает проблему циклических ссылок иначе — интеллектуальный алгоритм обнаруживает циклические ссылки, разрывает их и освобождает память.

Основные преимущества использования технологии .Net:

- платформенная независимость технологии .Net, легкость в разработке, встроенная система безопасности помогает создавать приложения, которые решают сложные деловые проблемы;
- создание приложений, использующих технологии, основанные на открытых стандартах, уменьшает риск и затраты на развертывание этих приложений и обеспечивает твердый фундамент для дальнейшего роста;
- технологии .Net, ASP .Net, ADO .Net обеспечивают общую платформу для различных устройств;
- использование приложений, основанных на платформе .Net, использующих преимущества многократно используемого, переносимого кода, помогает упростить развитие и сопровождение корпоративных систем;

- технологии .Net позволяют использовать инвестиции в ранее разработанные системы в следующем поколении приложений благодаря использованию межплатформенных возможностей. С технологиями .Net не существует такого понятия, как зависимость от поставщика;
- использование .Net-технологий позволяет компаниям двигаться быстрее, быть более гибкими и более эффективно реагировать на благоприятные условия рынка, создавая возможности для получения прибыли;
- по сравнению с возможностями технологии .Net на настольных компьютерах, серверах и мобильных устройствах, никакая другая технология не обладает таким признанием в промышленности и широтой распространения;

Основные возможности языка программирования VB.NET:

- принципы объектно-ориентированного программирования;
- расширенные возможности обработки исключительных ситуаций;
- богатый набор средств фильтрации ввода/вывода;
- встроенные простые классы, такие как массив, список, стек и т. п.;
- унифицированный доступ к базам данных на основе интерфейсов;
- Большая скорость разработки приложений.

Вышеперечисленные характеристики платформы .Net и языка программирования VB.NET послужили основной причиной при выборе его в качестве основного языка программирования, для написания системы морфологического анализа текста на узбекском языке. В работе были использованы широкие возможности работы с текстом, структурами, которые предоставляет VB.NET. Это позволило в значительной мере увеличить общую производительность работы системы анализа.

3.3. Разработка и реализация программных модулей проектирования базы данных морфологического словаря

Для поиска лемм в анализаторе используется два словаря: основной и дополнительный. В основном хранятся слова на узбекском языке, их перевод и указание на часть речи, к которой это слово относится. Дополнительный словарь используется для запоминания предсказанных слов. Этим достигается возможность “самообучения системы”

Ниже представлены структуры обеих таблиц.

Таблица LUGAT

Имя поля	Тип данных	Описание
ID	Счетчик	Код слова
UZB	Текстовый	Слово на узбекском
RUS	Текстовый	Перевод
TIP	Текстовый	Часть речи

Таблица AUTO

Имя поля	Тип данных	Описание
ID	Счетчик	Код слова
UZB	Текстовый	Слово на узбекском
TIP	Текстовый	Часть речи

В данной системе используется БД Microsoft Accsess, так как эта система является наиболее простой в использовании, при этом отвечает всем требованиям, необходимым для работы анализатора.

СТРУКТУРА АНАЛИЗАТОРА UZLUG

Основная задача данной диссертационной работы при построении сложной автоматизированной системы – это декомпозиция этой задачи на ряд более мелких и легких подзадач. Необходимо спроектировать систему, разбив ее

функционирование на ряд, логически обособленных друг от друга, модулей, что проиллюстрировано на рис.3.2.

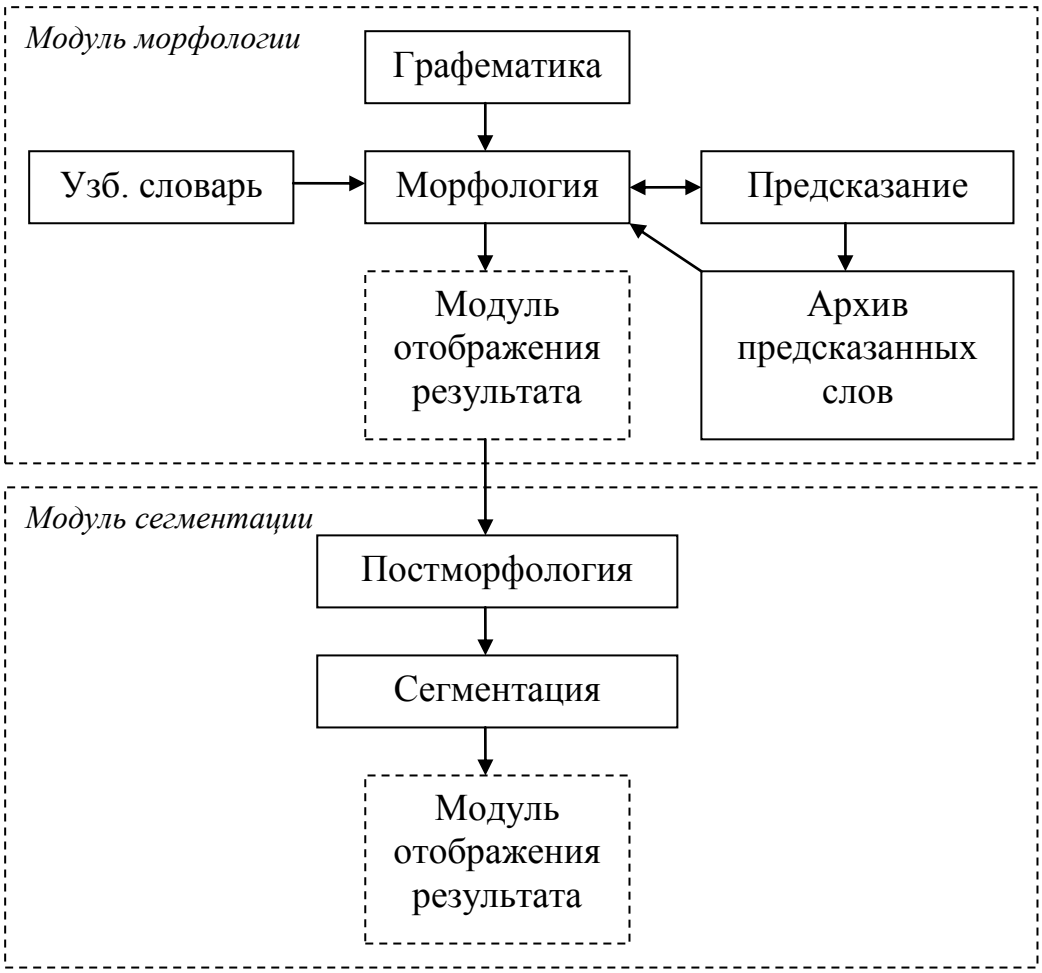

Рис. 3.2. Декомпозиция функциональных модулей анализатора словоформ

Под модулем системы в данной диссертационной работе понимается часть системы, которая имеет завершенный логический смысл, является логически обособленным от остальных узлов системы. На входе модуль системы получает определенный набор исходных данных, выполняет необходимую обработку и возвращает один набор результатных данных. Модульность в языках программирования – это принцип, согласно которому логически связанные между собой подпрограммы, переменные группируются в отдельные файлы (модули), которые компилируются независимо друг от друга. Модульность программы значительно уменьшает время её перекомпиляции при изменениях, вносимых в исходные тексты, упрощает групповую разработку.

На рисунке показана общая структура системы UzLUG. Система состоит из двух основных модулей: морфологического и сегментационного (синтаксического). В данной работе рассматривается модуль лемматизации и морфологического анализа.

Структура модуля морфологического анализа

Модуль морфологии состоит из трёх подмодулей: графематики, лемматизации и собственно модуля морфологического анализа (рис.3.3).

Модуль графематики постороен по упрощённой схеме. Входными данными являются предложения. На выходе модуль создаёт массив слов и знаков препинания. Все элементы в массиве имеют порядковый номер.

Модуль лемматизации состоит из нескольких процедур:

Процедура поиска устойчивых словосочетаний. Данная процедура последовательно объединяет 2-3 подряд идущих слова и вызывает процедуру поиска слов в словаре. Поиск идёт последовательным усечением словосочетания на один символ. Если словосочетание было найдено в словаре, то содержимое элементов массива объединяются.

Поиск слов в словаре. Поиск идёт последовательным усечением словосочетания на один символ. Для ускорения поиска применяется двоичный метод. При поиске слов учитываются словообразовательные префиксы. При этом часть речи у слова, найденная в слове будет зависеть от найденного префикса. Каждому найденному слову назначается “вес” равный единице.

Процедура предсказания слов. Все слова обрабатываются процедурой предсказывания основы. Это необходимо делать, так как слово найденное в словаре может быть омонимичным. Каждому предсказанному слову назначается “вес” равный нулю.

Все найденные леммы добавляются в массив слов с одинаковыми порядковыми номерами. На данном этапе может быть найдено большое количество ошибочных лемм.

Модуль морфологического анализа содержит процедуру обработки словообразовательных аффиксов, а также процедуры морфологического анализа для каждой части речи.

Процедуры обработки словообразовательных аффиксов и морфологического анализа могут как изменить значение части речи в массиве слов, так и добавить новый омоним. У новых омонимов снижается “вес”.

Рис. 3.3. Схема построения модуля морфологического анализа

Все процедуры в своей работе отталкиваются от найденной леммы. Анализируется часть начиная с последней буквы леммы и до конца всего слова. Необходимость учитывать последнюю буквы леммы продиктована тем, что в узбекском языке аффиксы могут меняться в зависимости от того, на какую букву оканчивается основа или предыдущий аффикс. Например, машина-м, китоб-им.

Модуль удаления ошибочных лемм работает следующим образом.

На начальном этапе процедура удаления ошибочных лемм проверяет все слова в массиве на правильность аффиксов. Проверку на правильность аффиксов осуществляет отдельный модуль. Данный модуль проверяет на возможность присутствия того или иного аффикса в определённой части речи. Проверка учитывает расположение аффиксов относительно друг друга. У всех слов, не прошедших проверку удаляется отметка о части речи и снижается “вес” до -1.

На втором этапе проверяются слова, отмеченные как омонимичные. Здесь удаляются слова, у которых “вес” меньше или нет отметки о указании части речи.

Для анализа результатов в систему включён модуль вывода результатов работы. Пример его работы показан на рис. 3.4.

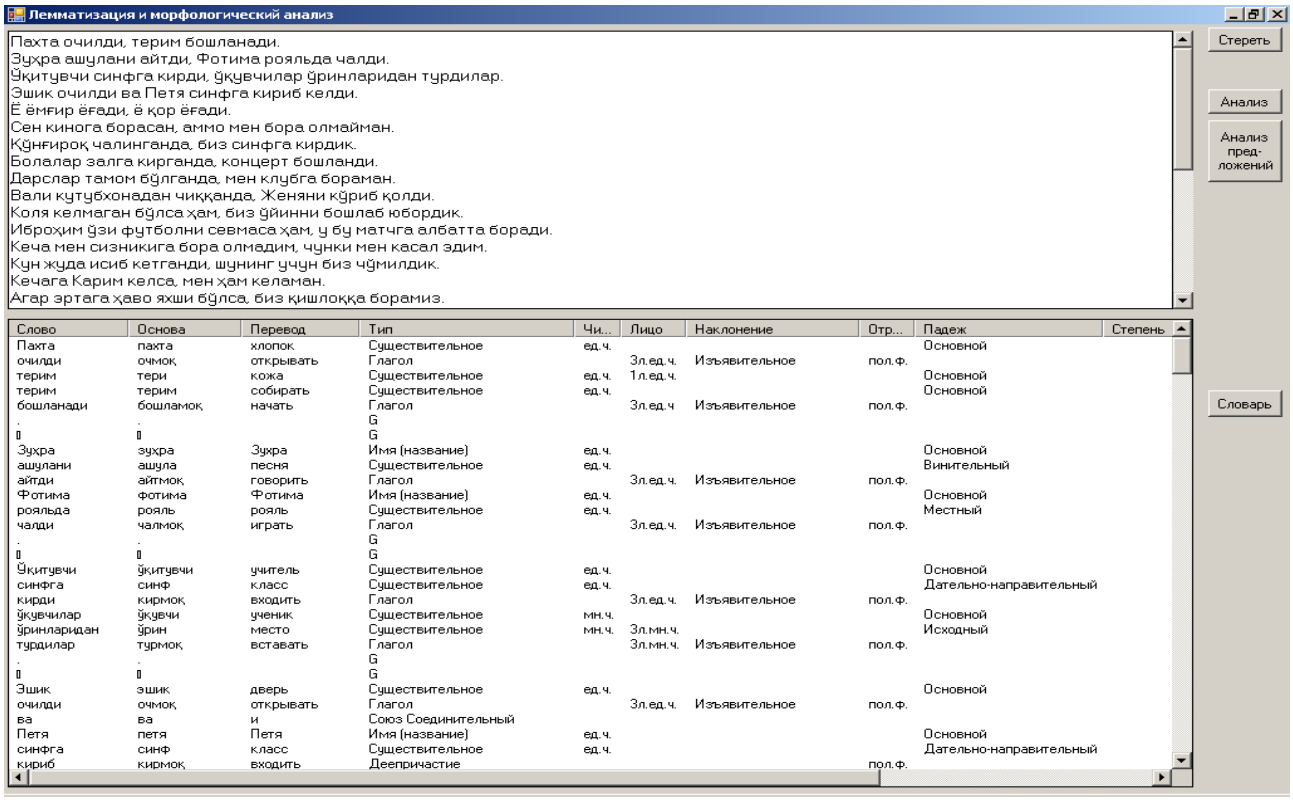

Рис. 3.4. Вид окна вывода результатов анализатора слваря словоформ.

Требования к аппаратному и программному обеспечению

Этап постановки задачи – один из наиболее ответственных этапов создания программного продукта. На этом этапе формулируют основные требования к разрабатываемому программному обеспечению. От того, насколько полно определены функции и эксплутационные требования, насколько правильно приняты принципиальные решения, определяющие процесс проектирования, во многом зависит стоимость разработки и ее качество.

Эксплутационные требования определяют некоторые характеристики разрабатываемого программного обеспечения, проявляемые в процессе его функционирования. К таким характеристикам относят:

- правильность – функционирования в соответствии с техническим заданием;
- универсальность – обеспечения правильной работы при любых допустимых данных и защиты от неправильных данных;
- надежность (помехозащищенность) – обеспечение полной повторяемости результатов, т. е. обеспечение их правильности при наличии различного рода сбоев;
- проверяемость – возможность проверки получаемых результатов;
- точность результатов – обеспечение погрешности результатов не выше заданной;
- защищенность – обеспечение конфиденциальности информации;
- программная совместимость – возможность совместного функционирования с некоторым оборудованием;
- эффективность – использование минимального возможного количества ресурсов технических средств, например, времени микропроцессора или объема оперативной памяти;
- адаптируемость – возможность быстрой модификации с целью приспособления к изменяющимся условиям функционирования;
- повторная входимость – возможность повторного выполнения без перезагрузки с диска;

- реентерабельность – возможность «параллельного» использования несколькими процессами.

Для работы системы необходимым условием является наличие соответственного программного и аппаратного обеспечения. Поскольку автоматизированная система написана на языке программирования VB.NET, то единственным условием работы является установленная среда исполнения – .Net Framework.

Данная система машинного анализа тестировалась на компьютере с конфигурацией: процессор – 2.6 Ггц, оперативная память – 256 Мб, монитор, клавиатура и мышь. Операционные системы: Windows XP и Windows Vista. Также возможна работа на ОС Windows 98 и семействе ОС Linux.

Примеры исходных данных и результатов обработки системы

Список исходных предложений:

- Пахта очилди, терим бошланади.
- Зуҳра ашулани айтди, Фотима рояльда чалди.
- Ўқитувчи синфга кирди, ўқувчилар ўринларидан турдилар.
- Эшик очилди ва Петя синфга кириб келди.
- Улар яхши ишлайдилар. (В словаре отсутствуют слова “ишламоқ” и “иш”)

Пахта пахта хлопок. Существительное ед.ч. Основной

Очилди очмоқ открывать. Глагол 3л.ед.ч. Изъявительное пол.ф. Очевидное прошедшее время Страдательный

Терим тери кожа. Существительное ед.ч. 1л.ед.ч. Основной
Терим терим собирать. Существительное ед.ч. Основной

Бошланади бошламоқ начать. Глагол 3л.ед.ч Изъявительное пол.ф. Настояще-будущее Страдательный

Зуҳра Зухра Имя (название) ед.ч. Основной

Ашулани ашула песня. Существительное ед.ч. Винительный

Айтди айтмоқ говорить. Глагол 3л.ед.ч. Изъявительное пол.ф. Очевидное прошедшее время

Фотима Фотима Имя (название) ед.ч. Основной

Рояльда рояль. Существительное ед.ч. Местный

Чалди чалмоқ играть. Глагол 3л.ед.ч. Изъявительное пол.ф. Очевидное прошедшее время

Ўқитувчи ўқитувчи учитель. Существительное ед.ч. Основной

Синфга синф класс Существительное ед.ч. Дательно-направительный

Кирди кирмоқ входить. Глагол 3л.ед.ч. Изъявительное пол.ф. Очевидное прошедшее время

Ўқувчилар ўқувчи ученик. Существительное мн.ч. Основной

Ўринларидан ўрин место. Существительное мн.ч. 3л.мн.ч. Исходный

Турдилар турмоқ вставать. Глагол 3л.мн.ч. Изъявительное пол.ф. Очевидное прошедшее время

Эшик эшик дверь Существительное ед.ч. Основной

Очилди очмоқ открывать Глагол 3л.ед.ч. Изъявительное пол.ф. Очевидное прошедшее время Страдательный

Ва ва и Союз Соединительный

Петя Петя Имя (название) ед.ч. Основной

Синфга синф класс Существительное ед.ч. Дательно-направительный

Кириб кирмоқ входить Деепричастие пол.ф. Прошедшее

Келди келмоқ приходить Глагол 3л.ед.ч. Изъявительное пол.ф. Очевидное прошедшее время

Улар у он. Местоимение. Личное мн.ч. 3л.ед.ч. Основной
Яхши яхши хорошо. Наречие меры и степени
Яхши яхши хороший. Существительное ед.ч. Основной

Ишлайдилар ишламоқ Глагол 3л.мн.ч. Изъявительное пол.ф. Настояще-будущее

Выводы по главе 3

1. Рассмотрены вопросы реализации алгоритмов контроля электронных текстов на узбекском языке. Предложен подход к созданию системы выделения лемм из электронных текстов на основе статистических данных. Наличие словаря лемм упрощает морфологический анализ слов, повышает точность. Каждой лемме в словаре возможно добавление перевода, что позволит построить машинный переводчик.

2. Разработана схема анализа текстов на узбекском языке для построения словаря словоформ, управление которым позволит упростить предсказание основы слова и его части речи на основе анализа аффиксов ввиду агглютинативности языка. Согласно схеме графематический модуль разбивает исходный текст документа на предложения и слова, выстраивая внешнее представление текста. Морфологический компонент осуществляет морфоанализ и лемматизацию узбекских словоформ.

3. Обоснован выбор программной среды для реализации системы контроля достоверности электронных текстов. Проанализированы основные возможности языка программирования VB.NET. Использованы широкие возможности работы с текстом, структурами, которые предоставляет VB.NET, что позволило в значительной мере увеличить общую производительность системы.

4. Разработаны и реализованы программные модули проектирования базы данных морфологического словаря. Реализована структура анализатора, которая имеет завершенный логический смысл, выполняет необходимую обработку и возвращает один набор результатов данных.

5. Определены функции и эксплутационные требования к программному обеспечению систем машинного анализа. На различных примерах исходных данных получены результаты обработки морологического лексикона.

ЗАКЛЮЧЕНИЕ

Результаты теоретических исследований и практических разработок, выполненных в диссертационной работе, сводятся к следующему:

1. Определены основные подходы к построению системы контроля и коррекции ошибок в текстах, которая основывается на использовании: статистики искажений; способов и моделей кодирования информации; программных методов контроля информации на основе искусственной и естественной избыточности; методов, моделей, алгоритмов морфологического анализа.

2. Разработаны методики: определения объема избыточности, обеспечивающего требуемое качество контроля текстов и расчета рационального объема памяти программной системы обработки информации для контроля и коррекции ошибок в текстах на естественных языках, в частности, на узбекском. Определено, что для обеспечения достоверности информации не менее 10^{-6} рекомендуемый объем избыточности должен быть не менее 0,4; объем рациональной памяти равен 2^{16} битов.

5. Исследованы правила описания формального анализа узбекских словоформ, на основе которой разработаны модель морфологического анализа, обобщенный алгоритм построения и структура программной системы для контроля и коррекции ошибок, не требующей от пользователя глубокого знания языка и позволяющей проводить контроль ошибок с ограниченным объемом словаря словоформ.

Получены следующие результаты решений частных задач:

- разработаны методы морфологического анализа применительно к узбекскому языку;
- созданы методы предсказания основы слова и части речи для существительных и глаголов;
- разработаны алгоритмы реализации и выполнено программирование системы на языке VB.NET;
- проведён анализ производительности системы, результаты которого удовлетворяют требованиям к качеству передачи и обработки информации.

ЛИТЕРАТУРА

Нормативно-правовые документы

1. Постановление Кабинета Министров Республики Узбекистан №150 от 10 апреля 1998г. «О создании межведомственной информационной компьютерной сети».
2. Распоряжение Кабинета Министров Республики Узбекистан №469 от 2 декабря 1994 г. «О концепции информатизации Республики Узбекистан».
3. Указ Президента Республики Узбекистан от 30 мая 2002 г. NУП-3080 «О дальнейшем развитии компьютеризации и внедрении информационно-коммуникационных технологий».
4. Постановление Кабинета Министров Республики Узбекистан №200 от 6 июня 2002 г. «О мерах по дальнейшему развитию компьютеризации и внедрению информационно-коммуникационных технологий».
5. «Положение о порядке и правилах создания, внедрения и эксплуатации локальных, ведомственных, региональных и других информационно-вычислительных сетей на территории Республики Узбекистан». Утверждено Госкомитетом Республики Узбекистан по науке и технике 30 января 1995г. (Зарегистрировано Министерством юстиции Республики Узбекистан №120 от 31 января 1995г.);
6. Каримов И.А. «Глобальный финансово-экономический кризис, пути и меры преодоления в условиях Узбекистана». – Т: Узбекистан, 2009, 56 с.

Основная литература

7. Абдурохмонов Ф., Сулаймонов А и др. Хозирги узбек адабий тили // изд. Ўқитувчи. Ташкент 1979. – с. 54-99.
8. Азларов Э. Рахимов А и др. Учебник узбекского языка // изд. Ўқитувчи. Ташкент 1993. – с.45-100.
9. Аношкина Ж.Г. Морфологический процессор русского языка. //Альманах «Говор», Сыктывкар, 1995, с.17-23.

10. Апресян Ю. Д., Богуславский И. М., Иомдин Л. Л., Лазурский А. В., Перцов Н. В., Санников В. З., Цинман Л. Л.. Лингвистическое обеспечение системы ЭТАП-2. - М.: Наука, 1989. – с. 65-87.
11. Ахатов А.Р., Химматов И.К. Алгоритмы контроля достоверности обработки электронных текстов в системах дистанционного обучения // «Современные информационно-комммуникационные технологии в образовании: проблемы и решения» илмий-амалий конференцияси материаллари, ТАТУ Самарқанд филиали, 2015 йил 15-16 апрел. – Самарқанд, 2015. - с.150-153.
12. Ахатов А.Р., Химматов И. Алгоритмы контроля достоверности обработки электронных текстов на основе оптического распознавания и управления морфологическим словарем. // «XXI аср – интеллектуал авлод асри» худудий илмий-амалий конференцияси, СамИСИ, Самарқанд, 2015 йил, 3-4 июнь, с. 236-241.
13. Ахатов А.Р., Химматов И.К. Технология контроля достоверности обработки электронных текстов в автоматизированных лингвистических системах // «Узлуксиз таълим сифат ва самарадорлигини оширишнинг назарий-услубий муаммолари» илмий конференция материаллари, Самарқанд, 24-25 ноябр 2015 йил. - СамДУ, 2015. – 95-98 б.
14. Бурлак С.А., Старостин С. А. Введение в лингвистическую компаративистику. – Эдиториал УРСС, М., 2001. – с.32-89.
15. Вирт Н. Алгоритмы и структуры данных. – СПб., 2001. – с. 35-150.
16. Г. Буч. Объектно-ориентированный анализ и проектирование. – М.: «Издательство Бином», 2000. - с. 25-75.
17. Грязнухина Т.А., Дарчук Н.П., Критская В.И., Маловица Н.П. и др. Синтаксический анализ научного текста на ЭВМ. - К.//Научная мысль, 1999. – с. 15-86.
18. Жуманов И.И., Джураев М.К. Метод коррекции текстов на основе вероятностной модели совершения ошибок. В РЖ «Вестник ТГТУ» № 1, Ташкент, 2004 г., с. 38-44.

19. Жуманов И.И., Джураев М.К. Использование статистики искажений в методах коррекции орфографических ошибок естественных языков. В РЖ «Вестник ТГТУ» № 2, 2004 г., Ташкент, с. 36-41.
20. Жуманов И.И., Джураев М.К. Коррекция орфографических ошибок с помощью системы оптического распознавания текстов. В сб. «Вопросы Кибернетики» № 169, Ташкент, 2004 г., с. 26-31.
21. Жураева Н.В. Разработка формальной модели грамматики узбекского языка и её программная реализация // VIII Всероссийская конференция молодых ученых по математическому моделированию и информационным технологиям. - Новосибирск., 2002. – с.34-38.
22. Ингве В. Гипотеза глубины. //Новое в лингвистике. Вып. 4, М., 1965 –126 с.
23. Касымова И.А. Разработка формальной модели и программы построения всех форм узбекского глагола и их соответствия в русском, английском языках // VIII Всероссийская конференция молодых ученых по математическому моделированию и информационным технологиям. Тезисы доклада. – Новосибирск, 2002. – с. 43-48.
24. Кобзарева Т.Ю., Лахути Д.Г., Ножов И.М. Сегментация русского предложения. // КИИ-2000. Труды конференции – М.: Физматлит, 2000. Т.1. - с. 339-344.
25. Кобзарева Т.Ю., Лахути Д.Г., Ножов И.М. Модель сегментации русского предложения. // Диалог’2001. Труды конференции – Аксаково, 2001. Т.2. - с. 185-194.
26. Кибрик А.Е. Очерки по общим и прикладным вопросам языкознания. – УРСС, М., 2001. – 145 с.
27. Кормалев Д.А., Куршев Е.П., Сулейманова Е.А., Трофимов И.В. Извлечение данных из текста. Анализ ситуаций ньюсмейкинга. // КИИ-2002. Труды конференции, т.1 – М., Физматлит, 2002. с. 56-64.
28. Мельчук И. Курс общей морфологии - Т.№1, М., 1997. – 211 с.
29. Мельчук И. А. Опыт теории лингвистических моделей “Смысл ⇒ Текст”. - М. «Наука», 1999. – 124 с.

30. Михаилян А. Некоторые методы автоматического анализа естественного языка, используемые в промышленных продуктах //www.citforum.ru/avtestla2ng.html . 2001
31. Ножов И.М. Синтаксический анализ. // Компьютерра, № 21 (446), 2002. – с. 34-37.
32. Ножов И.М. Прикладной морфологический анализ без словаря. // КИИ-2000. Труды конференции – М.: Физматлит, 2000. Т.1. - с. 424-429
33. Ножов И.М. Процессор автоматизированного морфологического анализа без словаря. Деревья и корреляция. //Диалог'2000. Труды конференции - Протвино, 2000. Т.2. - с. 284-290.
34. Ножов И.М. Проектирование сегментационного анализатора русского предложения. // КИИ-2002. Труды конференции – М.: Физматлит, 2002. Т.1. - с. 212-222.

Дополнительная литература

35. Сущанская Н. Ф. Программный препроцессор для естественноязыковых интерфейсов. - Автореф. дисс. к.т.н. – К.: РИО ИК, 1989. – 24 с.
36. Сокирко А. В. Семантические словари в автоматической обработке текста (по материалам системы Диалинг). - Автореф. дисс. к.т.н. – М., 2001. – 24 с.
37. Тестелец Я. Г. Введение в общий синтаксис. – М., РГГУ, 2001. – 125 с.
38. Ф. де Соссюр. Курс общей лингвистики. – М., 1999.- 145 с.
39. Федоров А. В. Основы общей теории перевода: Лингвистические проблемы. - М.: Высш.шк., 1968. - 303 с.
40. Харари Ф. Теория графов. - М., 1973.- 245 с.
41. Хамроев М.А. Ўзбек тилидан маърузалар тўплами. Ташкент 2003
42. Швейцер А.Д. К проблеме лингвистического изучения процесса перевода.//Вопросы языкознания. - 1970. -№ 4. - с.40-49.
43. Швейцер А.Д. Теория перевода. Статус, проблемы, аспекты – М.: Наука. 1998. – 215с.
44. Шереметьева С.О., Ниренбург С. Эмпирическое моделирование в вычислительной морфологии. //НТИ, №7, 1996. – с. 56-64.

45. Шевякова Д., Степанов А., Карпов А. Самоучитель Visual basic 2005. - СПб, 2006. – 452 с.
46. Шоабдурахмонов Ш, Аскарова М и др. Узбек тили стилистикаси // Укитувчи. Ташкент 1983. – 256 б.
47. Anthworth E. PC-KIMMO: A two-level processor for morphological analysis, Summer Institute of Linguistics, Texas, 1990.
48. J. K. Baker, “Stochastic modeling for automatic speech understanding,” in Speech Recognition, R. Reddy, Ed. New York: Academic Press, 1975, pp. 521–542.
49. Damerau F. J. A technique for computer detection and correction of spelling errors. Communications of the Association for Computing Machinary, 2010 7(3): pp.171-176, .
50. M. W. Du, S. C. Chang, A model and a fast algorithm for multiple errors spelling correction. Acta Informatica, no29, 1992. – pp..281-302.

Интернет сайты

51. www.aot.ru (Панкратов Д. В., Гершензон Л. М., Ножов И. М. Описание фрагментации и синтаксического анализа в системе Диалинг. // Техническая документация, , 2000)
52. www.osp.ru/os/2003/12/183694/_p2.html (Селезнев К. Обработка текстов на естественном языке)
53. www.library.ferghana.ru/uz/uzbsoz1.htm (Хамид Исмайлов. О философии узбекского языка)
54. http://www.analog.com processorsTussia/blackfm/index.html
55. http://www.analog.com-en epHSP.rod/0,2542,VISUALDSPBF,00.html
56. http://fastra.ua.ac.be/en/index.html
57. http://www.external.ameslab.gov/hoomd/index.html
58. http://www.springerlink.com/content/pk22n1632859082k/fulltext.html
59. http://www.linzik.com/

ПРИЛОЖЕНИЕ

```
Public Class FrmGap

    Public mor As New ArrayList
    Dim Gap As New ArrayList    '- предложения - вложеный аррейлист со словами типа frmmain.slov
    Dim gp As New ArrayList
    Dim segm As New ArrayList  ' - сегменты

    Private Structure Seg    '-структура сегмента
        Dim a As Integer
        Dim b As Integer
        Dim svz As String
        Dim Tip As String
        Dim Glv As Integer
        Dim urv As Integer
        Dim red As Boolean
        Dim Yes As Boolean
        Dim Im As Boolean
    End Structure

    Private Structure predl
        Dim slov As ArrayList
        Dim segm As ArrayList
    End Structure

    Sub PredSeg(ByRef mor As ArrayList)
        Dim t, t1, t2 As FrmMain.Slov
        Dim i As Integer = 0
        Dim n, omon, i2 As Integer
        For i = 0 To mor.Count - 1
            t = mor(i)
            t.Small = 0
            mor(i) = t
        Next
        i = 0
        Do Until i >= mor.Count - 1
            t = mor(i)
            t1 = mor(i + 1)
            If t.Omonim <> t1.Omonim Then
                If t.TIP = "C" AndAlso t.Padej = "п.п." Then
                    omon = t1.Omonim
                    n = omon
                    i2 = i + 1
                    Do Until n <> omon Or i2 > mor.Count - 1
                        If t1.TIP <> "C" Then
                            t1.Small = 1
                            mor(i2) = t1
                        End If
                        i2 = i2 + 1
                        t1 = mor(i2)
                        n = t2.Omonim
                    Loop
                    i = i2
                ElseIf t.TIP = "C" AndAlso t.Padej = "в.п." Then
                    omon = t1.Omonim
```

```
                n = omon
                i2 = i + 1
                Do Until n <> omon Or i2 > mor.Count - 1
                    If t1.TIP <> "Г" Then
                        t1.Small = 1
                        mor(i2) = t1
                    End If
                    i2 = i2 + 1
                    t1 = mor(i2)
                    n = t2.Omonim
                Loop
                i = i2

            ElseIf t.TIP = "С" AndAlso t.Padej = "о.п." Then
                If t1.TIP = "С" AndAlso t1.Padej = "о.п." Then
                    If i + 2 >= mor.Count Or (i + 2 < mor.Count AndAlso (mor(i + 2).omonim <> t1.Omonim) And
mor(i + 2).tip <> "С") Then
                        t.Padej = "п.п."
                        t.TEXT = t.TEXT & "(нинг)"
                        mor.Insert(i, t)
                        i = i + 1
                        '                       mor(i) = t
                    End If
                End If
            End If
        End If
        i = i + 1
    Loop
    Dim b2 As Boolean
    b2 = False
    Do Until b2
        b2 = True
        For i = mor.Count - 1 To 1 Step -1
            t = mor(i)
            t1 = mor(i - 1)
            If t.Omonim = t1.Omonim Then
                If t.Small > t1.Small Then
                    mor.RemoveAt(i)
                    b2 = False
                    Continue For
                ElseIf t.Small < t1.Small Then
                    mor.RemoveAt(i - 1)
                    b2 = False
                    Continue For
                End If
            End If
        Next
    Loop
End Sub

Sub run(ByRef mor As ArrayList)

    PredSeg(mor)

    Gap.Clear()
    Dim i As Integer
    i = -1
```

```
    Do Until i > mor.Count - 1
      i = FindGap(mor, i + 1)
    Loop
    ShowGap()
  End Sub


  Public Function FindGap(ByRef mor As ArrayList, ByVal i As Integer, Optional ByVal tp As ArrayList =
Nothing) As Integer
    Dim g, n As Integer
    g = 0
    n = 0
    Dim temp As New ArrayList
    If Not (tp Is Nothing) Then temp = tp.Clone
    If temp.Count > 0 Then temp.RemoveAt(temp.Count - 1)
    Dim b As Boolean = True
    Dim m As Integer = -1
    n = temp.Count - 1
    If i <= mor.Count - 1 Then
      Do Until mor(i).tip = "G"
        If Asc(mor(i).text) <= 13 Or mor(i).text = Nothing Then Continue Do
        If mor(i).omonim <> m Then
          temp.Add(mor(i))
          m = mor(i).omonim
          n = n + 1
        Else
          FindGap(mor, i, temp)
        End If
        i = i + 1
        If i > mor.Count - 1 Then Exit Do
      Loop
      Dim pr As New predl
      pr.slov = temp
      Dim s As New Seg
      s.a = 0
      s.b = n
      s.svz = ""
      segm = New ArrayList
      segm.Add(s)
      pr.segm = segm
      Gap.Add(pr)
    End If

    Return i
  End Function

  Private Sub FrmGap_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.Load
    run(mor)
  End Sub


  Sub ShowGap()
    LstGap.Items.Clear()
    Dim i, n As Integer
    Dim s As String = ""
    Dim s1 As String
    Dim g1 As predl
```

```
        For i = 0 To gap.Count - 1
            s = ""
            g1 = Gap(i)
            For n = 0 To g1.slov.Count - 1
                s1 = g1.slov(n).text
                If Asc(s1) <> 10 And Asc(s1) <> 13 Then
                    s = s & s1 & " "
                End If
            Next
            LstGap.Items.Add(s)
        Next
    End Sub

    Dim mash As Integer

    Private Sub LstGap_SelectedIndexChanged(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles LstGap.SelectedIndexChanged
        Dim i As Integer
        i = LstGap.SelectedIndex
        If i >= 0 AndAlso LstGap.SelectedItem.ToString.Length > 2 Then
            Dim g As Graphics
            g = Pict.CreateGraphics
            g.Clear(Color.White)
            Dim font As New Font("Courier New", 9, FontStyle.Regular)
            g.DrawString(LstGap.SelectedItem.ToString, font, Brushes.Black, 0, 150, New
StringFormat(Drawing.StringFormatFlags.NoClip))
            Dim n2 As Drawing.SizeF
            n2 = g.MeasureString(LstGap.SelectedItem.ToString, font, 10)
            mash = n2.Height \ LstGap.SelectedItem.ToString.Length
            Dim t As predl = Gap(LstGap.SelectedIndex)
            Dim s As String = ""
            Dim i1 As Integer
            For i1 = 0 To t.slov.Count - 1
                Dim n As FrmMain.Slov = t.slov(i1)
                If n.RUS.Length > 0 Then
                    s = s & n.RUS & " "
                Else
                    s = s & n.UZB & " "
                End If
            Next
            g.DrawString(s, font, Brushes.LightGray, 0, 195, New
StringFormat(Drawing.StringFormatFlags.NoClip))
            Analyse(i)
        Else
            Dim g As Graphics
            g = Pict.CreateGraphics
            g.Clear(Color.White)
        End If
    End Sub


    Function SegmGran(ByVal a As Integer, ByVal b As Integer, ByRef seg As ArrayList) As Integer
        Dim m As Integer
        For m = 0 To seg.Count - 1
            '  If seg(m).a = a AndAlso seg(m).b = b AndAlso seg(m).svz <> "" Then Return False
            If seg(m).yes = True AndAlso seg(m).a = a AndAlso seg(m).svz <> "" Then Return m
            If seg(m).yes = True AndAlso seg(m).b = a AndAlso seg(m).svz <> "" Then Return m
```

```
            If seg(m).yes = True AndAlso seg(m).a = b AndAlso seg(m).svz <> "" Then Return m
            If seg(m).yes = True AndAlso seg(m).b = b AndAlso seg(m).svz <> "" Then Return m
        Next
        Return -1
    End Function

    Function Podl_Skaz(ByRef n As Integer, ByRef ts As ArrayList, ByRef seg As ArrayList, ByRef b1 As
Boolean) As Boolean
        Dim a, b As Integer
        If n < seg.Count Then
            If seg(n).tip = "Г" AndAlso seg(n).yes = True Then
                Dim m As Integer
                For m = n - 1 To 0 Step -1
                    If seg(m).yes = True AndAlso seg(m).im Then
                        ' If seg(m).tip = "ЮС" Or seg(m).tip = "Z" Then Exit For
                        'Dim t As String = seg(m).tip
                        b = seg(n).glv
                        a = seg(m).glv
                        Dim k1, k2 As String
                        k1 = ts(b).litco & " "
                        k2 = ts(a).litco & " "
                        If (k1.Substring(0, 1) <> "3" AndAlso k1 = k2) Or (k1.Substring(0, 1) = "3" AndAlso k2 = " ")
Or (k1.Substring(0, 1) = "3" AndAlso k1.Substring(0, 1) = k2.Substring(0, 1)) Then
                            Dim sg1 As New Seg
                            sg1.b = b
                            sg1.svz = "ПОДЛ_СКАЗ"
                            sg1.Tip = "W"
                            sg1.Glv = b
                            sg1.urv = 0
                            Dim v As Integer
                            For v = m To n
                                If seg(v).urv > sg1.urv Then sg1.urv = seg(v).urv
                            Next
                            sg1.urv = sg1.urv + 1
                            sg1.red = False
                            sg1.Yes = True
                            seg.Insert(n + 1, sg1)
                            Return True
                        End If

                    End If
                Next
            End If

        End If

        Return b1
    End Function


    Function Chis_Such(ByRef n As Integer, ByRef ts As ArrayList, ByRef seg As ArrayList, ByRef b1 As
Boolean) As Boolean
        Dim a, b As Integer
        If n < seg.Count - 1 Then
            If seg(n).tip.ToString.Length > 0 AndAlso seg(n).tip.ToString.Substring(0, 1) = "Ч" AndAlso seg(n).yes
= True Then
                Dim m As Integer
```

```
            For m = n + 1 To seg.Count - 1
               If seg(m).yes = True Then Exit For
            Next
            Dim t As String = seg(m).tip
            If t = "C" AndAlso seg(m).yes = True Then
               a = seg(n).glv
               b = seg(m).glv
               sg1.b = b
               sg1.svz = "ЧИСЛ_СУЩ"
               sg1.Tip = t
               sg1.Glv = b
               sg1.Im = seg(m).im
               sg1.urv = IIf(seg(n).urv > seg(m).urv, seg(n).urv + 1, seg(m).urv + 1)
               sg1.red = False
               sg1.Yes = True
               seg.Insert(m + 1, sg1)
               Return True
            End If
         End If
      End If
      Return b1
   End Function


   Function Pril_Such(ByRef n As Integer, ByRef ts As ArrayList, ByRef seg As ArrayList, ByRef b1 As
Boolean) As Boolean
      Dim a, b As Integer
      If n < seg.Count - 1 Then
         If seg(n).tip = "П" AndAlso seg(n).yes = True Then
            Dim m As Integer
            For m = n + 1 To seg.Count - 1
               If seg(m).yes = True Then Exit For
            Next
            If m = seg.Count Then Return b1
            If seg(m).tip = "С" Then
               a = seg(n).glv
               sg1.a = a
               sg1.b = b
               sg1.svz = "ПРИЛ-СУЩ"
               sg1.Tip = "С"
               sg1.Glv = b
               sg1.Im = seg(m).im
               sg1.urv = IIf(seg(n).urv > seg(m).urv, seg(n).urv + 1, seg(m).urv + 1)
               sg1.red = False
               sg1.Yes = True
               seg.Insert(m + 1, sg1)
               Return True
            End If
         End If
      End If
      Return b1
   End Function

   Function Prid_Vremya(ByRef n As Integer, ByRef ts As ArrayList, ByRef seg As ArrayList, ByRef b1 As
Boolean) As Boolean
      Dim a, b As Integer
      If n < seg.Count - 2 Then
```

```
        If seg(n).tip = "Z" AndAlso seg(n).yes = True Then
            Dim m As Integer
            For m = n - 1 To 0 Step -1
                If seg(m).yes = True Then Exit For
            Next
            If m < 0 Then Return b1
            Dim m1 As Integer
            For m1 = n + 1 To seg.Count - 1
                If seg(m1).yes = True Then Exit For
            Next
            If m1 = seg.Count Then Return b1
            Dim t As FrmMain.Slov = ts(seg(m).b)
            If seg(m1).tip = "W" AndAlso t.TIP = "K" AndAlso t.Padej = "м.п." AndAlso t.Vremya = "Пр.в."
Then
                a = seg(m).a
                sg1 = New Seg
                sg1.a = a
                sg1.b = b
                sg1.svz = "СЛОЖ.ПОДЧ.ВРЕМ."
                sg1.Tip = "W"
                sg1.Glv = b
                sg1.urv = IIf(seg(m).urv > seg(m1).urv, seg(m).urv + 1, seg(m1).urv + 1)
                sg1.red = True
                sg1.Yes = True
                seg.Insert(m1 + 1, sg1)
                Return True
            End If
        End If
    End If
    Return b1
End Function


Function Vremya(ByRef n As Integer, ByRef ts As ArrayList, ByRef seg As ArrayList, ByRef b1 As
Boolean) As Boolean
    Dim a, b As Integer
    If seg(n).tip = "C" AndAlso seg(n).yes = True Then
        If n < seg.Count - 1 Then
            If ts(seg(n).glv).text = "соат" Then
                Dim m As Integer
                For m = n + 1 To seg.Count - 1
                    If seg(m).yes = True Then Exit For
                Next
                If seg(m).tip = "ЧК" AndAlso seg(m).yes = True Then
                    a = seg(n).glv
                    b = seg(m).glv
                    ' If SegmGran(a, b, seg) = -1 Then
                    Dim sg1 As New Seg
                    sg1 = seg(n)
                    sg1.Yes = False
                    seg(n) = sg1
                    sg1 = New Seg
                    sg1 = seg(m)
                    sg1.Yes = False
                    seg(n + 1) = sg1
                    sg1.a = a
                    sg1.b = b
```

```
                sg1.svz = "ВРЕМЯ"
                sg1.Tip = "Н"
                sg1.Glv = b
                sg1.urv = IIf(seg(n).urv > seg(m).urv, seg(n).urv + 1, seg(m).urv + 1)
                sg1.red = True
                sg1.Yes = True
                seg.Insert(m + 1, sg1)
                Return True
              End If
            End If
          End If
        End If
        Return b1
    End Function



    Sub Analyse(ByRef ind As Integer)
        Dim pred As predl = Gap(ind)
        Dim g As Graphics
        g = Pict.CreateGraphics

        Dim i As Integer
        Dim seg As New ArrayList
        Dim ts As New ArrayList
        ts = pred.slov
        seg = pred.segm
        seg.Clear()

        Dim z As Boolean = False
        Dim t As New FrmMain.Slov

        For i = 0 To ts.Count - 1
            t = ts(i)
            Dim s As New Seg
            s.a = i
            s.b = i
            s.svz = ""
            s.urv = 0
            s.Tip = t.TIP
            If (t.TIP = "С" And t.Padej = "о.п.") Or (t.TIP = "И" And (t.Padej = "о.п." Or t.Padej = "")) Or t.TIP =
"МЛ" Or t.TIP = "МО" Or t.TIP = "МД" Then
                s.Im = True
            Else
                s.Im = False
            End If
            s.red = False
            s.Glv = i
            If s.Tip = "В" Then
                s.Yes = False
            Else
                s.Yes = True
            End If

            seg.Add(s)
        Next
        For i = 0 To seg.Count - 1
```

```
    Dim s As New Seg
    s = seg(i)
    If s.Tip = "B" Then
        If i > 0 Then
            Dim s1 As New Seg
            s1 = seg(i - 1)
            If s1.Tip = "Z" Then
                s1.Yes = False
                seg(i - 1) = s1
            End If
        End If
        If i < seg.Count - 1 Then
            Dim s1 As New Seg
            s1 = seg(i + 1)
            If s1.Tip = "Z" Then
                s1.Yes = False
                seg(i + 1) = s1
            End If
        End If
    End If
Next
Dim b1 As Boolean = True
Dim sg As New Seg
Dim stamp As New ArrayList

Do Until Not (b1)
    b1 = False
    For i = 0 To seg.Count - 1
        b1 = Odnor_Pril(i, ts, seg, b1)
    Next
Loop
b1 = True
Do Until Not (b1)
    b1 = False
    For i = 0 To seg.Count - 1
        b1 = Pril_Such(i, ts, seg, b1)
    Next
Loop
b1 = True
Do Until Not (b1)
    b1 = False
    For i = 0 To seg.Count - 1
        b1 = GENIAT_PARA(i, ts, seg, b1)
    Next
Loop
b1 = True
Do Until Not (b1)
    b1 = False
    For i = 0 To seg.Count - 1
        b1 = PR_Such(i, ts, seg, b1)
    Next
Loop
b1 = True
Do Until Not (b1)
    b1 = False
    For i = 0 To seg.Count - 1
        b1 = Odnor_Nar(i, ts, seg, b1)
```

```
    Next
Loop
b1 = True
Do Until Not (b1)
    b1 = False
    For i = 0 To seg.Count - 1
        b1 = Kolich(i, ts, seg, b1)
    Next
Loop
b1 = True
Do Until Not (b1)
    b1 = False
    For i = 0 To seg.Count - 1
        b1 = Sl_Glag(i, ts, seg, b1)
    Next
Loop
b1 = True
Do Until Not (b1)
    b1 = False
    For i = 0 To seg.Count - 1
        b1 = Nar_Prich(i, ts, seg, b1)
    Next
Loop
b1 = True
Do Until Not (b1)
    b1 = False
    For i = 0 To seg.Count - 1
        b1 = Nar_Glag(i, ts, seg, b1)
    Next
Loop
b1 = True
Do Until Not (b1)
    b1 = False
    For i = 0 To seg.Count - 1
        b1 = FIO(i, ts, seg, b1)
    Next
Loop
b1 = True
Do Until Not (b1)
    b1 = False
    For i = 0 To seg.Count - 1
        b1 = Nar_Chis_Such(i, ts, seg, b1)
    Next
Loop
b1 = True
Do Until Not (b1)
    b1 = False
    For i = 0 To seg.Count - 1
        b1 = Chis_Such(i, ts, seg, b1)
    Next
Loop

b1 = True
Do Until Not (b1)
    b1 = False
    For i = 0 To seg.Count - 1
        b1 = Deep_Prich(i, ts, seg, b1)
```

```
   Next
Loop

b1 = True
Do Until Not (b1)
   b1 = False
   For i = 0 To seg.Count - 1
      b1 = Mest_Such(i, ts, seg, b1)
   Next
Loop


b1 = True
Do Until Not (b1)
   b1 = False
   For i = 0 To seg.Count - 1
      b1 = Odnor_Such(i, ts, seg, b1)
   Next
Loop
b1 = True
Do Until Not (b1)
   b1 = False
   For i = 0 To seg.Count - 1
      b1 = Vremya(i, ts, seg, b1)
   Next
Loop
b1 = True
Do Until Not (b1)
   b1 = False
   For i = 0 To seg.Count - 1
      b1 = PRCH(i, ts, seg, b1)
   Next
Loop
b1 = True
Do Until Not (b1)
   b1 = False
   For i = 0 To seg.Count - 1
      b1 = Such_PRICH(i, ts, seg, b1)
   Next
Loop
b1 = True
Do Until Not (b1)
   b1 = False
   For i = 0 To seg.Count - 1
      b1 = Praym_Dopol(i, ts, seg, b1)
   Next
Loop
b1 = True
Do Until Not (b1)
   b1 = False
   For i = 0 To seg.Count - 1
      b1 = Podl_Skaz(i, ts, seg, b1)
   Next
Loop

b1 = True
Do Until Not (b1)
```

```
        b1 = False
        For i = 0 To seg.Count - 1
            b1 = Nar_Glag(i, ts, seg, b1)
        Next
    Loop
    b1 = True
    Do Until Not (b1)
        b1 = False
        For i = 0 To seg.Count - 1
            b1 = Sloj_Soch(i, ts, seg, b1)
            b1 = Prid_Vremya(i, ts, seg, b1)
            b1 = Prid_Ustup(i, ts, seg, b1)
            b1 = Prid_Prich(i, ts, seg, b1)
            b1 = Prid_Sel(i, ts, seg, b1)
            b1 = Prid_Mesta(i, ts, seg, b1)
        Next
    Loop
    printseg(ind)
    prin(ind)
End Sub
End Class
```

О Т З Ы В

на диссертационную работу магистранта Химматова Ибодилла «Разработка алгоритмов и программных средств контроля достоверности обработки электронных текстов на основе методов управления морфологическим словарем» по специальности 5А110701 – Информационные технологии в образовании.

Применение любой информационной системы (ИС) управления связано с появлением ошибок в передаваемой, вводимой и обрабатываемой информации. Источниками возникновения ошибок в составе информации являются человек-оператор ИС, процессы сканирования и распознавания, помехи в каналах связи и ограниченная надежность технических средств подготовки данных и технических носителей информации, орфографические ошибки в машинописных текстах.

Следовательно, весьма актуальной является проблема исследования и разработки методов повышения достоверности передачи, ввода и обработки информации, как важного фактора обеспечения надежного и качественного функционирования систем переработки информации.

Существующие методы не охватывают большую часть ошибок, которые допускаются человеком-оператором при нанесении информации на машинные носители и при вводе их в память ЭВМ, ошибок сканирования и распознавания, орфографических ошибок. Практика показывает, что на долю человека-оператора как раз приходится примерно 85% ошибок из общего объема искажения, а погрешность существующих систем сканирования и распознавания примерно равна 10^{-2}. Причем требуется обеспечить достоверность информации до 10^{-5}-10^{-7}.

Известные методы могут быть удачно дополнены программными методами контроля достоверности информации, использующими искусственную и статистическую (естественную) избыточность информации.

Большинство программных методов контроля основаны на использовании вводимой избыточности данных, реализуются на ЭВМ, используются в основном на практике обработки цифровых данных и дают наилучший эффект при обнаружении ошибок.

Доказано, что в ИС делопроизводства предприятий и организаций, наряду с цифровой информацией, обрабатывается большое количество служебных документов и машинописные тексты, обладающие значительной естественной избыточностью. Причем, использование вложенных в тексты логических и статистических связей символов и букв, семантики и свойств естественного языка, как раз и создает благоприятные условия для обеспечения качества передачи, ввода и обработки машинописной информации, в том числе коррекции орфографических ошибок.

Однако, в работах, посвященных использованию статистической избыточности информации, теоретические и практические стороны поставленной нами проблемы контроля ошибок изучены пока недостаточно. Отсутствуют приемлемые методы, алгоритмы и правила контроля ошибок человека-оператора, сканирования и распознавания и орфографических ошибок, особенно в машинописных текстах на узбекском языке.

Следовательно, решение этой проблемы требует теоретического обоснования и проведения специальных исследований и разработок, а с точки зрения практики переработки информации необходимо устранение существующих недостатков в известных методах, обеспечение требуемой достоверности информации при незначительных временных и материальных затратах, реализация их в составе ИС без привлечения дополнительной техники.

В связи с этим в диссертационной работе исследуется и решается актуальная научно-техническая задача, связанная с разработкой технологии повышения качества передачи, ввода и обработки машинописных текстов на узбекском, русском и английском языках.

Целью исследования является исследование и разработка научно-методических основ, способов и алгоритмов контроля и исправления орфографических ошибок узбекского языка на основе метода перебора и управления морфологическим словарем на этапах их подготовки, ввода, передачи и обработки, а также практическое использование результатов исследований в системах автоматизации делопроизводства.

В соответствии с целью работы решены следующие теоретические и практические задачи:

- изучены лингвистические стратегии и правила, отвечающие словообразовательным законам языка, проанализировано применение описываемых грамматических конструкций в корпусе текстов.
- разработаны способы морфологического анализа без словаря и методики анализа словоформ в морфологической модели;
- разработаны алгоритмы морфологического анализа на основе индексирования и проектирование словарной морфологии;
- обоснована сложность машинного анализа узбекского предложения, разработана методика постморфологического анализа;
- разработана общая схема действий сегментационного анализа и внутрисегментного анализа;
- предложена архитектура и программно-реализуемые модули системы контроля и исправления орфографических ошибок на основе моделей управления морфологическим словарем.

В заключении можно отметить, что в диссертационной работе решена актуальная научная задача, имеющая большое народно-хозяйственное значение. Работа выполнена на высоком теоретическом уровне. Диссертационная работа соответствует предъявляемым требованиям, а ее автор И.Химматов заслуживает присуждения степени магистра по специальности 5А110701 – Информационные технологии в образовании.

Научный руководитель,
профессор кафедры «Информационных технологий»
СамГУ, д.т.н. А.Р.Ахатов

Р Е Ц Е Н З И Я

на диссертационную работу магистранта Химматова Ибодилла «Разработка алгоритмов и программных средств контроля достоверности обработки электронных текстов на основе методов управления морфологическим словарем» по специальности 5А110701 – Информационные технологии в образовании.

В информационных системах передаются для обработки цифровые, аналого-дискретные, графические и текстовые информации. Передача любой информации связана с появлением в ней ошибок. Статистика показывает, что в существующих условиях качество перерабатываемой информации равна примерно 10^{-3}, а требуется повысить достоверность информации на 2-3 порядка.

В связи с этим в диссертационной работе поставлено решение актуальной задачи, связанной с исследованием, разработкой методов, моделей, алгоритмов и программных средств для повышения качества передачи текстовой информации, а также практического использования результатов в информационных системах.

В частности решены следующие теоретические и практические задачи по теме выбранного направления исследований:

- определены основные подходы к построению программной системы контроля и коррекции ошибок в текстах, которая основывается на использовании: статистики искажений; методов, моделей, алгоритмов морфологического анализа.

- разработаны методики определения объема избыточности, обеспечивающего требуемое качество контроля текстов и расчета рационального объема памяти программной системы обработки информации для контроля и коррекции ошибок в текстах на естественных языках, в частности, на узбекском. Определено, что для обеспечения достоверности информации не менее 10^{-6} рекомендуемый объем избыточности должен быть не менее 0,4; объем рациональной памяти равен 2^{16} битов.

Исследованы правила описания формального анализа узбекских словоформ, на основе которой разработаны модель морфологического анализа, обобщенный алгоритм построения и структура программной системы для контроля и коррекции ошибок, не требующей от пользователя глубокого знания языка и позволяющей проводить контроль ошибок с ограниченным объемом словаря словоформ.

В заключении можно отметить, что в диссертационной работе решена актуальная научная задача, имеющая большое народно-хозяйственное значение. Работа выполнена на высоком теоретическом уровне. Диссертационная работа соответствует предъявляемым требованиям, а ее автор И.Химматов заслуживает присуждения степени магистра по специальности 5А110701 – Информационные технологии в образовани.

Заместитель директора
по научной работе Самаркандского
филиала ТУИТ, к.т.н., доц.: Бекмурадов К.А.

Р Е Ц Е Н З И Я

на диссертационную работу магистранта Химматова Ибодилла «Разработка алгоритмов и программных средств контроля достоверности обработки электронных текстов на основе методов управления морфологическим словарем» по специальности 5А110701 – Информационные технологии в образовании.

Для эффективного функционирования систем электронного документооборота требуется обеспечить достоверность информации на 2-3 порядка выше существующих показателей, что является актуальной задачей и имеет большое теоретическое и практическое значение. Одним из базовых принципов обеспечения достоверности информации является использование информационной избыточности, как искусственной, так и естественной.

Наиболее важным этапом обработки информации с точки зрения обеспечения достоверности является ввод информации в систему. Именно на этапе ввода должны быть исключены ошибки информации, вводимой в систему для обработки. Возможность решения задачи выявления ошибок при вводе информации может обеспечиваться как стандартными средствами систем управления базами данных - сохранение целостности доменов базы данных по ограничениям на диапазон допустимых значений, а также специально разработанными комплексами программных средств.

В связи с этим в диссертационной работе поставлено решение актуальной задачи, связанной с исследованием, разработкой методов, моделей, алгоритмов и программных средств для повышения качества передачи текстовой информации, а также практического использования результатов в информационных системах.

Разработаны методики определения объема избыточности, обеспечивающего требуемое качество контроля текстов и расчета рационального объема памяти программной системы обработки информации для контроля и коррекции ошибок в текстах на естественных языках, в частности, на узбекском.

Исследованы правила описания формального анализа узбекских словоформ, на основе которой разработаны модель морфологического анализа, обобщенный алгоритм построения и структура программной системы для контроля и коррекции ошибок, не требующей от пользователя глубокого знания языка и позволяющей проводить контроль ошибок с ограниченным объемом словаря словоформ.

В заключении можно отметить, что в диссертационной работе решена актуальная научная задача, имеющая большое народно-хозяйственное значение. Работа выполнена на высоком теоретическом уровне. Диссертационная работа соответствует предъявляемым требованиям, а ее автор И.Химматов заслуживает присуждения степени магистра по специальности 5А110701 – Информационные технологии в образовании.

Заведующий кафедры «Информационных технологий»
СамГУ, к.т.н., доц.: Джуманов О.И.

МЕХАНИКА-МАТЕМАТИКА ФАКУЛЬТЕТИ
АХБОРОТЛАШТИРИШ ТЕХНОЛОГИЯЛАРИ КАФЕДРАСИ

мажлис баёнидан

К У Ч И Р М А

10-2016 й 23.05.2016

Катнашдилар: проф. Жуманов И.И.- семинар раиси; доц. Джуманов О.И. - кафедра мудири; Кобилов С.С-доцент; Туракулов И.-доцент; Ахатов А.Р.- доцент; Очилов Т., - доцент; Аминов И.Б.- доцент, Абдуллаев А.Н.- доцент; Очилов С. - доцент; Бустонов Х. - катта укитувчи; ва бошка укитувчилар. Жами: 15 киши.

Кун тартиби:

1.Магистрант И.Қ.Химматов"Разработка методов и алгоритмов построения системы контроля и исправления орфографических ошибок узбекского языка на основе способов управления морфологическим словарем" магистрлик диссертациясини мухокама килиш ва уни химояга тавсия килиш.

Сузга чикди проф. И.И.Жуманов. Магистрант Ибодилла Химматов 2014 йили СамДУ Механика-математика факультетини “ Амалий математика ва информатика” мутахассислиги буйича "бакалавр" унвони билан «аъло» бахоларга тамомлаган. 2016 йили Механика-математика факультети ахборотлаштириш технологиялари кафедрасининг 5А110701 - "Таълимда ахборот технологиялари " мутахассислиги буйича магистратурага кирган.

Магистратурада утилган фанларни "аъло ва яхши" бахоларга топширган. Факультет рахбариятининг ижобий тавсифномасига эга. 2016 йили " Разработка методов и алгоритмов построения системы контроля и исправления орфографических ошибок узбекского языка на основе способов управления морфологическим словарем" мавзусидаги магистрлик диссертациясининг мавзуси тасдикланган.

Илмий рахбар техника фанлари номзоди, доцент А.Р.Ахатов.

Магистрант И.Қ.Химматов диссертацияси мухокамага куйилади. Суз магистрант И.Химматовга берилди.

Сузга чикди магистрант И.Қ.Химматов. Магистрант диссертация мавзусининг долзарблигини, диссертациянинг максадини, диссертацияда куйилган масалаларни, диссертациянинг асосий мазмунини ва ишлаб чикилган хулоса ва тавсияларни изохлади.

Диссертация мазмуни буйича семинар иштирокчиларидан проф. И.И.Жуманов, доц. Джуманов О.И., доц. Абдуллаевлар саволлар беришди.

Магистрант берилган саволларга коникарли ва тула жавоб берди.
Диссертация буйича илмий рахбар доц. А.Р.Ахатов сўзга чикди.

Сузга чикувчилар диссертация мавзусининг долзарблигини, диссертация иши назарий ва амалий жихатдан катта ахамиятга эга булган илмий амалий муаммонинг ечимига багишланганлигини таъкидлашди. Магистрант диссертацияда куйилган масалаларни муваффакиятли ечган. Бу эса магистрант И.Қ.Химматовнинг утилган фанларни чукур узлаштирганлигини курсатади. Магистрлик диссертацияси Давлат Аттестацияси Комиссиясига химоя килиш учун тавсия этилади.

Барча семинар иштирокчилари доц. А.Р.Ахатовнинг фикр мулохазаларига кушилишди ва магистрант И.Қ.Химматовнинг диссертациясини химоя килишга тавсия этишди.

КАРОР: 1.Магистрант И.Қ.Химматовнинг "Разработка методов и алгоритмов построения системы контроля и исправления орфографических ошибок узбекского языка на основе способов управления морфологическим словарем" мавзусидаги диссертацияси тамомланган илмий иш мавкеига эга ва назарий ва амалий жихатдан катта кизикиш билдирадиган илмий масала ечимига багишланган.

2.Магистрант И.Қ.Химматовнинг диссертацияси буйича берилган илмий рахбарнинг ижобий мулохазаси, расмий ва оппонентнинг такризи тасдиклансин.

3.Магистрант И.Қ.Химматовнинг диссертацияси барча куйилган талабларга жавоб беради ва Давлат Аттестация Комиссиясига химоя килишга тавсия этилади.

Илмий семинар раиси,
Ахболротлаштириш технологиялари
кафедраси профессори, фан.доктори И.И.Жуманов

Илмий котиб, тех.фан.
номзоди доц:

5А 110701 “Таълимда ахборот технологиялари“ мутахассислиги буйича магистрант И.Химматовга

Т А В С И Ф Н О М А

Химматов Ибодилла Қудратович 1 август 1989 йилда Қўшработ туманида тугилган. 2007 йили СамҚХИ қошидаги 2-сон Академик лицейни томомлаган.

СамДУ “Механика-математика” факультетига “ Амалий математика ва информатика” мутахассислиги буйича укишга кирган ва уни 2014 йили “бакалавр” унвони билан тамомлаган.

2014 йили СамДУ “Механика-математика” факультети, “Ахборотлаштириш технологиялари” кафедрасининг 5А110701 “Таълимда ахборот технологиялари” мутахассислиги буйича магистратурага кирган ва магистратурада утилган фанларни, илмий амалиетни, илмий-педагогик амалиетни “яхши” ва “аъло” бахоларга топширган.

Магистрант И.Химматов магистратурада укиш пайтида маънавий ва маърифий ишлар билан боглик жамоат ишлар билан боглик жамоат тадбирларига фаол кат нашди ва факультет рахбарияти топшириклари ни бажариб келди.

Магистрант И.Химматов намунали хулқ-атворга ва талабалар жамоаси олдида яхши обруга эга.

Факультет декани: доц. Х.Х.Рузимурадов

Кафедра мудири: доц. О.И.Джуманов