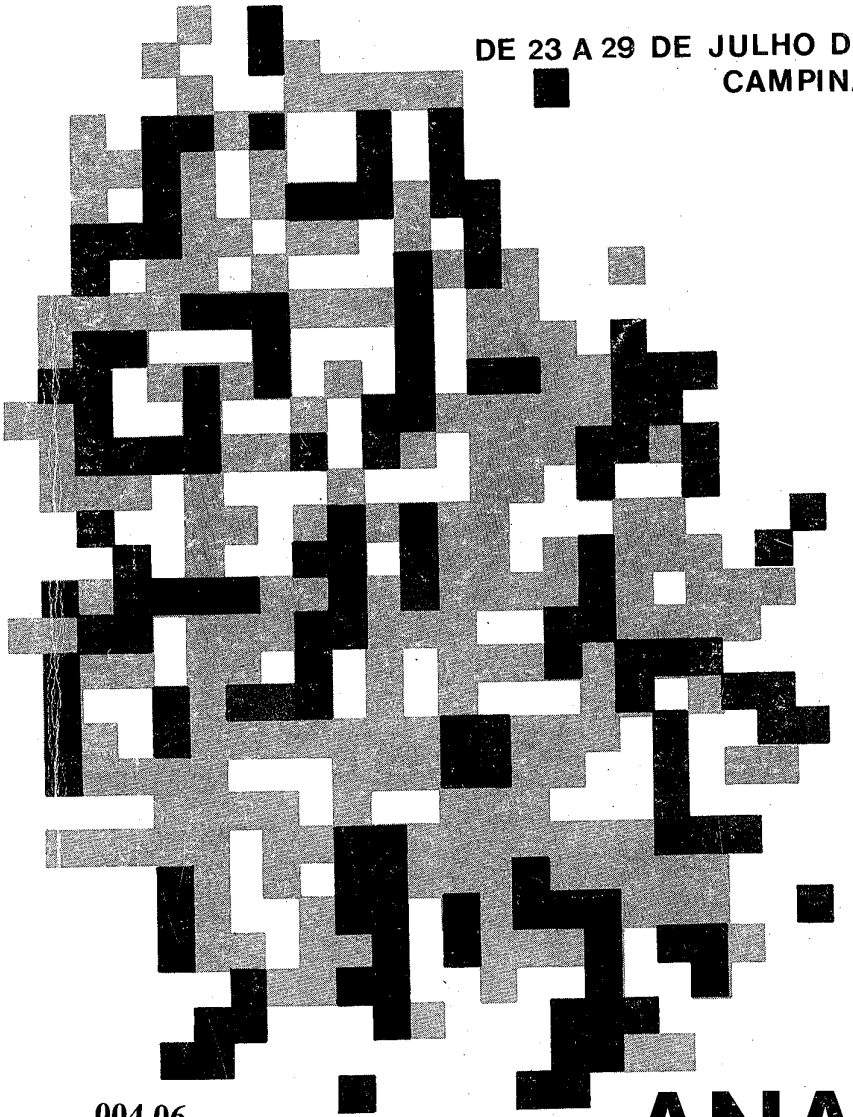
III CONGRESSO

DA SOCIEDADE BRASILEIRA DE COMPUTAÇÃO

DE 23 A 29 DE JULHO DE 1983
CAMPINAS - SP



ANAIS

VOL. I

III CONGRESSO DA SOCIEDADE BRASILEIRA DE COMPUTAÇÃO
CAMPINAS 23 a 29 de julho de 1983

ANAIS
VOLUME I

TRABALHOS APRESENTADOS
X SEMINÁRIO INTEGRADO DE SOFTWARE E HARDWARE

EDITORES: N. MEISEL, L.J. BRAGA-FILHO

PROMOÇÃO: SOCIEDADE BRASILEIRA DE COMPUTAÇÃO - SBC
UNIVERSIDADE ESTADUAL DE CAMPINAS - UNICAMP

PATROCÍNIO: CAPES, CNPq, DIGIBRÁS, FINEP, SEI

CO-PATROCINADORES: CPqD/TELEBRAS, PUC-Campinas, UFV
PREFEITURA MUNICIPAL DE CAMPINAS

P R I X P A L : UM INDEXADOR SEMI AUTOMÁTICO

A. von STAA

SUMÁRIO

Ao redigir livros didáticos, documentação de sistemas, etc. é usual requerer-se a criação de índices remissivos (referências cruzadas) facilitando ao leitor encontrar as porções do texto que possam vir a responder perguntas específicas. A construção de tais índices é tediosa, demorada e custosa. Além disto, em ambientes onde sejam frequentes as alterações, induz a uma considerável perda de tempo. Deseja-se, pois, automatizar a geração de índices remissivos.

Neste artigo é descrito um sistema adaptativo de geração de índices remissivos associados ao formatador de textos ATF disponível na PUC/RJ. O sistema foi projetado para a criação de índices de documentos em língua portuguesa auxiliando tanto na seleção de verbetes como na incorporação de marcadores no texto fonte.

* Engenheiro Mecânico (PUC 1965), Mestre em Informática (PUC/RJ, 1969) e Doutor em Ciência da Computação (Universidade de Waterloo, Canadá 1974). É Professor Associado no Departamento de Informática da PUC/RJ. Áreas de interesse compreendem: Engenharia de Software e Ferramentas para o Apoio no Desenvolvimento e à Documentação de Software. Departamento de Informática PUC/RJ; Rua Marques de S. Vicente, 225; 22453 - Rio de Janeiro - Tel. 274-9922 R. 386.

Este trabalho foi realizado com apoio da FINEP.

INTRODUÇÃO

Índices remissivos visam facilitar a procura de tópicos específicos tratados nos respectivos documentos. Diversos documentos devem possuir índices remissivos, entre eles livros técnicos, documentação de sistemas, etc. A criação de índices remissivos é tediosa e demorada, requerendo de quem indexa um bom domínio do assunto. Sendo assim a indexação deveria ser efetuada uma única vez durante toda a vida útil do documento.

Muitos documentos têm vida longa, sendo modificados com alguma frequência durante sua vida útil, por exemplo manuais de sistemas. Muitos destes documentos são hoje produzidos utilizando processadores de palavras ou formatadores de texto. Surge então, imediatamente, a idéia de automatizar-se a geração de índices remissivos. O assunto central deste artigo é apresentar o sistema de auxílio à preparação de índices remissivos PRIXPAL desenvolvido para operar junto com o formatador de textos ATF [Staa 78] disponível na PUC/RJ.

Um dos princípios norteadores do sistema PRIXPAL foi o de procurar dar soluções adaptativas, ao invés de dar uma solução geral pré-estabelecida. Assim, utilizando um sistema interativo, o usuário instrui o sistema a gerar índices de acordo com os seus desejos. Desta forma o sistema é mantido bem simples e, apesar disto, é eficaz e eficiente. O preço pago é um esforço inicial de instrução do sistema de indexação a ser despendido para cada documento a indexar. Este esforço é, no entanto, pequeno e tende a diminuir à medida que o sistema se torne melhor instruído.

O sistema PRIXPAL foi desenvolvido em SPITBOL ("threaded interpreter" do SNOBOL), consta de 9 programas (totalizando cerca de 500 linhas de comando), utiliza 7 arquivos e foi utilizado para gerar o índice remissivo do livro [Staa 83].

CRIAÇÃO DO TEXTO DO ÍNDICE

A primeira pergunta que surge é como automatizar a preparação do texto do índice. É claro que o formatador deve ser empregado para tal. É claro, também, que a geração das referências aos números de página tem que ser efetuada pelo formatador pois ele é o responsável pela diagramação e paginação do texto formatado (saída do formatador). Devido às ambiguidades e sensibilidades ao contexto inerentes a qualquer linguagem natural, fica difícil criar um sistema baseado somente no texto. Assim sendo conclui-se que o texto fonte (entrada ao formatador) deve conter comandos de marcação indicando a ocorrência, naquele local, de um tópico a ser referenciado no índice. Adotando-se esta solução, o texto fonte passará a ter o formato:

texto do corpo // texto do índice

onde texto do corpo contém o texto propriamente dito, comandos de formatação e comandos de marcação. Texto do índice contém as palavras do índice e os comandos de incorporação

das listas de número de páginas marcadas.

Para simplificar o processo de marcação, utilizou-se uma tabela de símbolos. Cada tópico no índice está associado a um símbolo exclusivo. Este símbolo é utilizado pelos comandos de marcação e de incorporação. O comando de marcação gera a lista de números de página onde o verbete ou noção é usado. O comando de incorporação obtém a lista associada ao símbolo e a fornece como texto fonte ao formatador.

A maioria dos sistemas comerciais de processamento de textos e que dão suporte à indexação param aqui. Com este instrumental mínimo já é possível criar-se índices remissivos adaptáveis às alterações de texto. No entanto, continua-se necessitando de uma pessoa experiente para criar o texto do índice e introduzir os comandos de marcação.

O problema, agora, passou a ser: como incorporar os comandos de marcação e como gerar o texto fonte do índice contendo os verbetes e os respectivos comandos de incorporação da listas de referências. É evidente que se deseja automatizar estas duas atividades. A solução adotada foi a de utilizar um dicionário de verbetes, onde cada verbete (ou frase) está relacionado a um único tópico. De posse deste dicionário, o texto fonte é percorrido e são gerados os comandos de marcação. Para tornar mais geral a aplicabilidade do programa, os verbetes encontrados no texto fonte são identificados no dicionário de modo que o índice remissivo gerado contenha os verbetes para os quais exista no mínimo uma referência no texto fonte original. Pode-se, assim, utilizar um dicionário referenciando mais tópicos do que o texto a ser indexado. A figura 1 mostra o fluxo do sub-sistema de marcação.

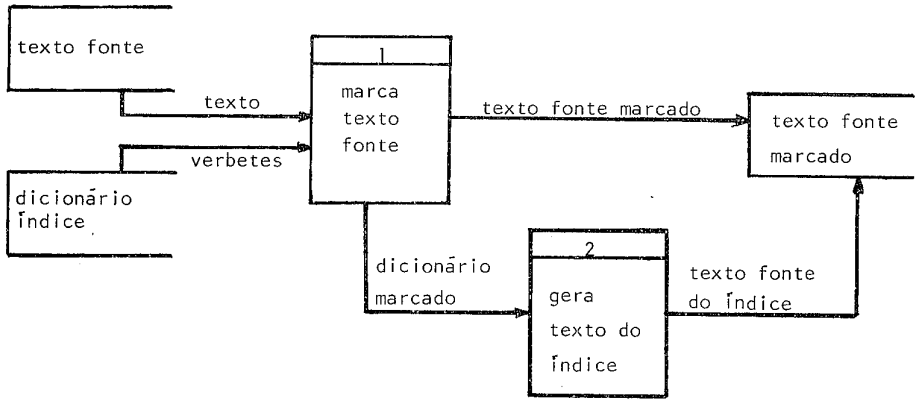


Figura 1. Fluxo do sub-sistema de marcação

Para reduzir o tamanho do dicionário foram eliminadas as terminações de gênero e número dos diferentes verbetes do índice, guardando-se somente os prefixos destes verbetes. Para acelerar o processo de marcação o dicionário de prefixos é convertido para um autômato finito. Para cada palavra do texto fonte a marcar será ativado este autômato. A palavra será descartada caso o autômato atinja um estado de impasse (i.e. o autômato encontra-se em um estado para o qual não existe transição com o caractere corrente). Será gerado um comando de marcação caso o autômato atinja um estado final válido. Cada estado final corresponde a um único prefixo de verbete. A construção do autômato controla possíveis ambiguidades. Cada prefixo de verbete está relacionado a um tópico a ser incluído no texto do índice, onde o texto descritivo deste tópico, tal como aparece no texto do índice remissivo, pode ser diferente do verbete encontrado. Por exemplo, ao encontrar o prefixo "galh" (prefixo dos verbetes "galho" de "galhos") pode ser disparada a marcação e inclusão do tópico "manual de quebra galhos". Ou seja, o dicionário de prefixos associa a cada prefixo o texto que deverá ser incluído no índice remissivo caso o prefixo em questão ocorra em uma das palavras do texto fonte a ser marcado.

Existe uma pequena possibilidade de erro, uma vez que determinada palavra pode satisfazer o prefixo e no entanto não corresponder ao respectivo tópico de interesse. Por exemplo, "cama" e "camada" satisfazem o prefixo "cam" correspondente ao verbete "cama", porém ao marcar "camada" estará sendo cometido um erro de indexação. Foram conduzidos vários experimentos com textos extensos (250 páginas impresas, 130K palavras) e não foi constatado qualquer caso de incorreção de marcação. Ou seja, teoricamente o algorítimo pode errar, porém na prática tais erros são muito pouco frequentes. Assim sendo é pouco justificável o custo de cerca de 30% a mais na tabela do autômato reconhecedor para chegar a um algorítmo quase infalível. Este algorítmo requer um dicionário contendo os textos completos dos verbetes a indexar, sendo que são gerados comandos de marcação se e somente se a palavra inteira corresponde a um verbete do dicionário. O algorítmo é quase-infalível, pois falhará caso nos interesse apenas uma das possíveis acepções da palavra. É evidente que isto somente será possível examinando-se em mais detalhe e contexto onde se encontra o uso da palavra. Por exemplo, a palavra "nós" pode interessar enquanto significar um nó em uma rede de teleprocessamento, não interessando quando for pronome. Note que em uma frase tal como "A REDPUC é feita por nós", qualquer sistema mecanizado terá que recorrer ao entendimento do contexto da frase para poder determinar o significado da palavra "nós".

Este processo de marcação automático tem a vantagem de permitir a posterior inclusão e/ou exclusão de comandos de marcação efetuada por meios manuais utilizando um sistema interativo de edição. Assim possíveis erros e/ou dificuldades encontrados poderão sempre ser corrigidos por intervenção direta. O que se deseja é que o volume de correção seja pequeno, menor do que 1%. Nos experimentos conduzidos, não foi sentida a necessidade da correção manual.

O problema agora transformou-se em como criar o dicionário de verbetes de índice. O dicionário deve conter verbetes simples, expressões compostas e indicadores de noções de inte-

resse para o índice. Com relação a noções, a automação foi abandonada uma vez que é necessário a extração do sentido do texto para poder marcá-lo. Por exemplo, o presente texto referencia a noção: "adaptabilidade através do aprendizado", onde esta noção deveria ser marcada em todas as páginas que tratem da criação e/ou manutenção de arquivos. Expressões compostas (ex. qualidade de software, qualidade de texto etc.) envolvem a criação de algoritmos capazes de identificar padrões complexos onde um ou mais termos podem estar sendo fornecidos por contexto. Apesar de reconhecidamente importantes (grande parte dos índices são expressões compostas), seu tratamento foi deixado para o futuro, uma vez que se desejava era mostrar que é possivelmente válida a conjectura "o uso de um sistema adaptativo para criar índices remissivos é eficaz e eficiente".

CRIAÇÃO DO DICIONÁRIO DE VERBETES

A pergunta, agora, é como identificar quais são os verbetes a figurarem no índice. Qualquer solução envolve a leitura do texto inteiro e a eliminação ou filtragem das palavras que não interessem. Como grande parte deste trabalho pode ser realizado mecanicamente, optou-se por um processo interativo de ensino ao computador. As hipóteses de trabalho foram 1) o número de iterações de ensino é pequeno, 2) o esforço de ensino por iteração é pequeno tendendo assintóticamente a zero à medida que o sistema vai aprendendo. A figura 2 mostra em linhas gerais o fluxo do processo de aprendizado.

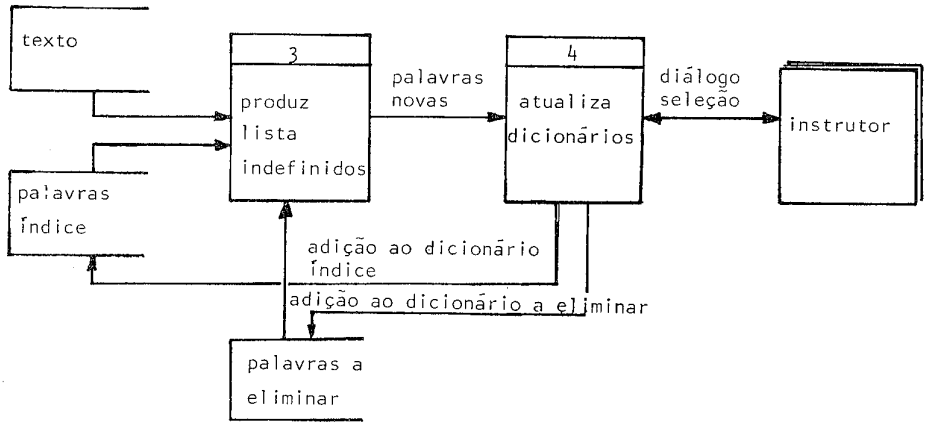


Figura 2 Fluxo do processo de aprendizado

O processo de aprendizado utiliza 3 filtros. O primeiro elimina todas as repetições de palavras e todas as palavras auxiliares tais como advérbios, pronomes, conjunções, verbos auxiliares etc. O resultado desta primeira filtragem é um texto contendo uma única ocorrência de cada palavra existente no texto dado e formado quase exclusivamente por verbos, adjetivos e substantivos. As palavras após este primeiro filtro ainda estão completas, ou seja, mantêm as diferentes terminações de gênero, número, grau e conjunção . Tendo em vista certas homografias e a dificuldade para criar um dicionário completo, algumas palavras que deveriam ter sido eliminadas neste primeiro processo de filtragem poderão ter passado para a próxima etapa. O dicionário utilizado na primeira filtragem é bastante estável e independe do redator e do assunto de que trata o texto a ser indexado. Ou seja, o aprendizado esperado, com relação a este dicionário é virtualmente zero.

Examinando-se a língua portuguesa constata-se a existência de várias palavras diferentes, divergindo somente na terminação sem divergir no seu significado intrínseco, uma vez que o português é particularmente rico em terminações verbais e de gênero, número e grau. Para reduzir o esforço de seleção de palavras é desejável agrupar palavras de mesmo sentido embora de diferentes terminações. Idealmente ao construir estes agrupamentos, as palavras deveriam ser agrupadas conforme sejam substantivos, adjetivos ou verbos. No entanto, a prática mostrou que esta separação não é primordial para a eficácia do sistema.

A primeira solução adotada foi a de eliminar as terminações verbais, de gênero, número e grau. A porção inicial da palavra (prefixo) serve então como símbolo em uma tabela de símbolos. Note que este prefixo é diferente (em geral mais curto) do prefixo utilizado para a marcação do texto fonte. A cada prefixo é associada uma lista contendo as palavras que passaram pelo primeiro filtro e para as quais o algoritmo de supressão de terminações gera o mesmo prefixo. O resultado foi bem melhor do que o esperado sendo que a maior causa de problemas foram os sufixos aumentativos e diminutivos. Eliminaram-se estes sufixos e restringiram-se as terminações apenas às porções variáveis, reduzindo-se, assim, a cardinalidade dos agrupamentos. Por exemplo, ao invés de utilizar as terminações "ção" e "ções", utilizou-se somente "ão" "ões" uma vez que "ç" está presente nas duas terminações. Exemplos de agrupamentos de palavras:

desenvolv	: desenvolvidos desenvolvido desenvolvidas desenvolver desenvolvendo
desenvolvime	: desenvolvimento
eleme	: elemento elementos
element	: elementar elementares

Note que alguns dos prefixos podem ser prefixos de outros existentes no conjunto. No entanto tal não representa fonte de ambiguidade, uma vez que a identificação de prefixo a partir de um conjunto de palavras fornecidas é determinístico e repetitivo.

Em mais de 99% dos casos os agrupamentos vieram separados de modo satisfatório. Entende-se por separação satisfatória uma separação em que todas as palavras do agrupamento correspondem a um mesmo significado e onde, se uma palavra do agrupamento não deva ser marcada, então todas as palavras deste agrupamento não deverão ser marcadas. Nos nossos exemplos existiram alguns (2) agrupamentos contendo palavras de significado diferente, porém não ocorreu caso de agrupamento misturando palavras a indexar com outras.

Tendo em vista o alto grau de satisfação, esta solução foi adotada deixando-se para o instrutor a tarefa de corrigir os poucos casos de agrupamentos não satisfatórios. Isto tornou possível criar um algorítmo que separa o conjunto de agrupamentos em 3 classes:

1. agrupamento eliminados: segundo o instrutor, todas as palavras contidas neste agrupamento jamais ocorrerão em índices remissivos de textos redigidos pelo mesmo autor sobre determinado assunto. A maioria das formas verbais e dos adjetivos cai nesta categoria (filtro 2).

2. agrupamentos incluidos: segundo o instrutor, todas as palavras destes agrupamentos deverão ser mercados no presente texto (filtro 3). Em geral esta lista contém somente substantivos.

3. agrupamentos indefinidos, que contém as demais palavras. Uma observação interessante é que erros de grafia são apontados nesta lista, em geral como prefixos isolados, facilitando assim a correção ortográfica do texto.

Para separar as listas foram criados dois dicionários, um de prefixos a eliminar e outro de prefixos do índice. Ao proceder assim verificou-se que o dicionário de prefixos a eliminar contém uma parte "constante" para área de interesse do autor, a "constante do autor". O restante do dicionário de prefixos a eliminar depende do texto a ser indexado. O dicionário de prefixos de índice deve ser criado integralmente para cada texto a indexar.

O processo de criação dos dois dicionários é portanto:

1. inicializar o dicionário de Prefixos a eliminar com a "constante do autor"

2. obter a lista de agrupamentos indefinidos. Esta lista tende a ser pequena (cerca de 200 prefixos e 600 palavras no nosso texto exemplo).

3. utilizando um sistema interativo, o instrutor separa manualmente a lista de agrupamentos indefinidos em, i) uma lista de prefixos a eliminar já existente; ii) uma lista de prefixos de índices e, iii) o resto permanece na lista de agrupamentos indefinidos.

4. com este novo conjunto de dicionários (eliminar e índice) repete-se o procedimen

to (passos 2, 3 e 4) até que a lista de agrupamentos indefinidos esteja vazia ou contenha apenas palavras que não deverão ser marcadas.

Verificou-se que sendo-se parcimonioso na inclusão de prefixos no dicionário de prefixos a eliminar, este dicionário correspondia à "constante do autor", podendo ser utilizado como inicializador do arquivo de prefixos tal como descrito no passo 1. Verificou-se, também, que quanto maior a constante do autor menor o esforço de seleção de prefixos a serem incluídos no dicionário de índice. Deve-se, no entanto, tomar o cuidado de não incluir no dicionário de prefixos a eliminar (constante do autor) prefixos que potencialmente poderiam, mais tarde, vir a interessar como índice, ou que têm existência efêmera, por exemplo nomes de autores, erros de ortografia, etc.

A criação do autômato de marcação é possível, agora, utilizando-se as listas de palavras agrupadas aos prefixos contidos no dicionário de prefixos de índice. Note que no caso de necessitar-se efetuar uma correção por meio manual, este dicionário deverá ser corrigido por intermédio de um editor interativo. Como anteriormente observado, os prefixos de índice são diferentes (menores) do que os prefixos de marcação.

A figura 3 mostra o fluxo do subsistema de auxílio à criação do dicionário de verbetes do índice.

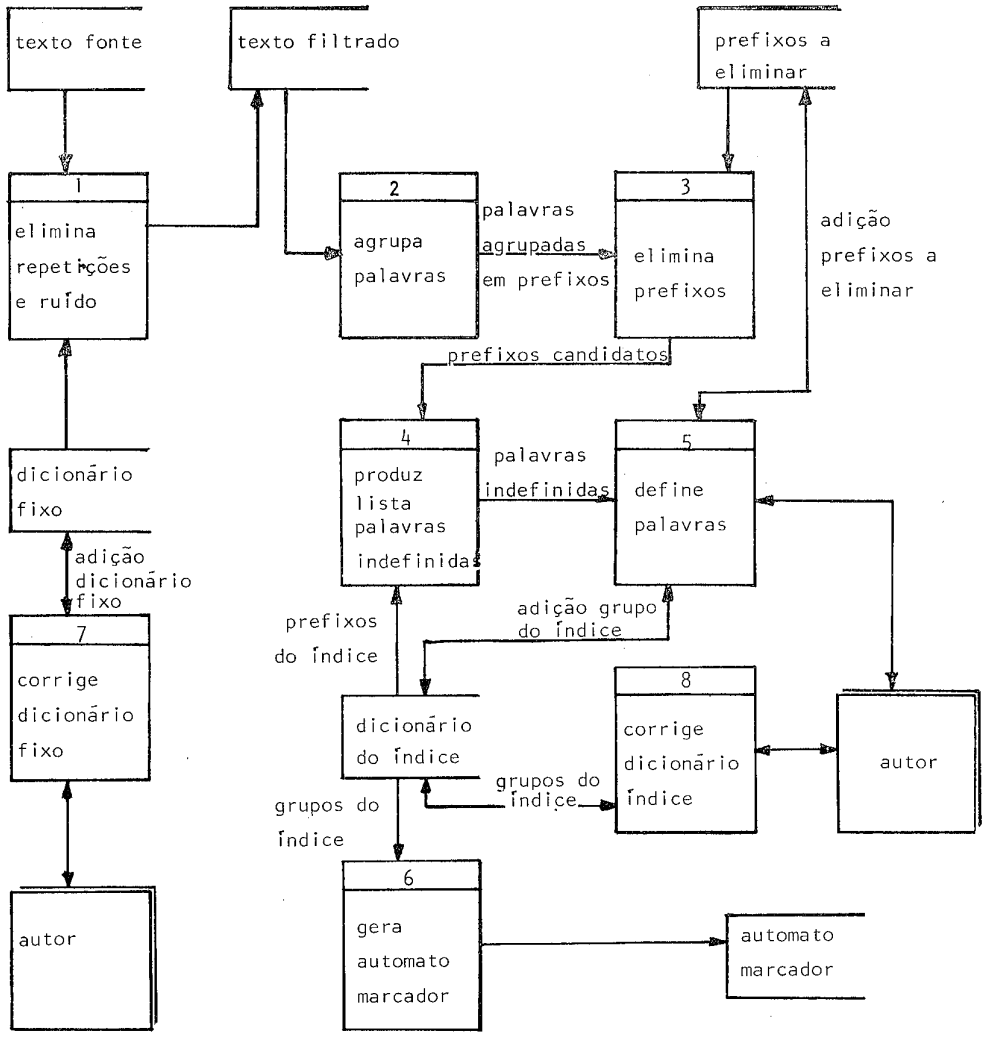


Figura 3 Fluxo do sub-sistema de apoio à criação do dicionário de verbetes do índice.

CONCLUSÃO

O presente sistema foi utilizado para gerar o índice remissivo de um livro a ser publicado comercialmente. A "instrução" do sistema foi efetuada pelo autor em cerca de 5 horas. Cabe mencionar que, somente para ler 250 páginas, o indexador utilizaria mais do que 5 horas.

Além disto, o número de erros observados foi muito pequeno, considerando como erros: excesso e/ou falta de palavras no índice, e listas de referência incompletas. Assim sendo, julgamos que o sistema satisfaz os requisitos de eficácia e eficiência.

O exercício sustentou também a hipótese de que sistemas adaptáveis aos usuários podem ser soluções adequadas, apesar do esforço inicial necessário para adaptá-los às suas necessidades específicas e apesar da taxa de falha que possam ter. É claro que, se o ambiente de execução não tivesse sido interativo, este processo de adaptação teria sido muito mais penoso e demorado. Ou seja, para uma determinada classe de problemas resolvidos em ambiente interativo, em muitas ocasiões é melhor dar uma solução adaptável e quase correta, fornecendo um mecanismo de correção, do que tentar uma solução geral e perfeita. Cabe observar ainda que uma grande parcela dos problemas reais não permitem soluções gerais nem perfeitas, uma vez que não dispomos da base teórica necessária e/ou por ser a imperfeição inerente ao problema a resolver.

REFERÊNCIAS

[Staa 78] Staa, A.V.
ATF - Formatador de Textos, documentação de sistema; "on-line" no RDC-PUC/RJ; Departamento de Informática; PUC/RJ; 1978

[Staa 83] Staa, A.V.
Engenharia de Programas; Livros Técnicos e Científicos; Rio de Janeiro; 1983.